The Engineering Staff of
TEXAS INSTRUMENTS LIMITED
Semiconductor Group

TMS 9900 Microprocessor Data Manual

August 1978

TEXAS INSTRUMENTS
LIMITED

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF ILLUSTRATIONS

LIST OF TABLES

1. INTRODUCTION

1.1 DESCRIPTION

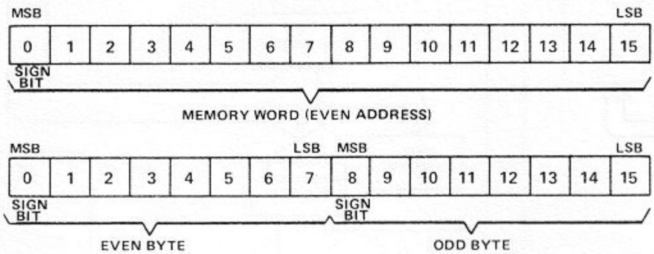
The TMS 9900 microprocessor is a single-chip 16-bit central processing unit (CPU) produced using N-channel silicon-gate MOS technology (see Figure 1). The instruction set of the TMS 9900 includes the capabilities offered by full minicomputers. The unique memory-to-memory architecture features multiple register files, resident in memory, which allow faster response to interrupts and increased programming flexibility. The separate bus structure simplifies the system design effort. Texas Instruments provides a compatible set of MOS and TTL memory and logic function circuits to be used with a TMS 9900 system. The system is fully supported by software and a complete prototyping system.

1.2 KEY FEATURES

- 16-Bit Instruction Word
- Full Minicomputer Instruction Set Capability Including Multiply and Divide
- Up to 65,536 Bytes of Memory
- 3-MHz Speed
- Advanced Memory-to-Memory Architecture
- Separate Memory, I/O, and Interrupt-Bus Structures
- 16 General Registers
- 16 Prioritized Interrupts
- Programmed and DMA I/O Capability
- N-Channel Silicon-Gate Technology

2. ARCHITECTURE

The memory word of the TMS 9900 is 16 bits long. Each word is also defined as 2 bytes of 8 bits. The instruction set of the TMS 9900 allows both word and byte operands. Thus, all memory locations are on even address boundaries and byte instructions can address either the even or odd byte. The memory space is 65,536 bytes or 32,768 words. The word and byte formats are shown below.



2.1 REGISTERS AND MEMORY

The TMS 9900 employs an advanced memory-to-memory architecture. Blocks of memory designated as workspace replace internal-hardware registers with program-data registers. The TMS 9900 memory map is shown in Figure 2. The first 32 words are used for interrupt trap vectors. The next contiguous block of 32 memory words is used by the extended operation (XOP) instruction for trap vectors. The last two memory words, $FFFC_{16}$ and $FFFE_{16}$, are used for the trap vector of the LOAD signal. The remaining memory is then available for programs, data, and workspace registers. If desired, any of the special areas may also be used as general memory.

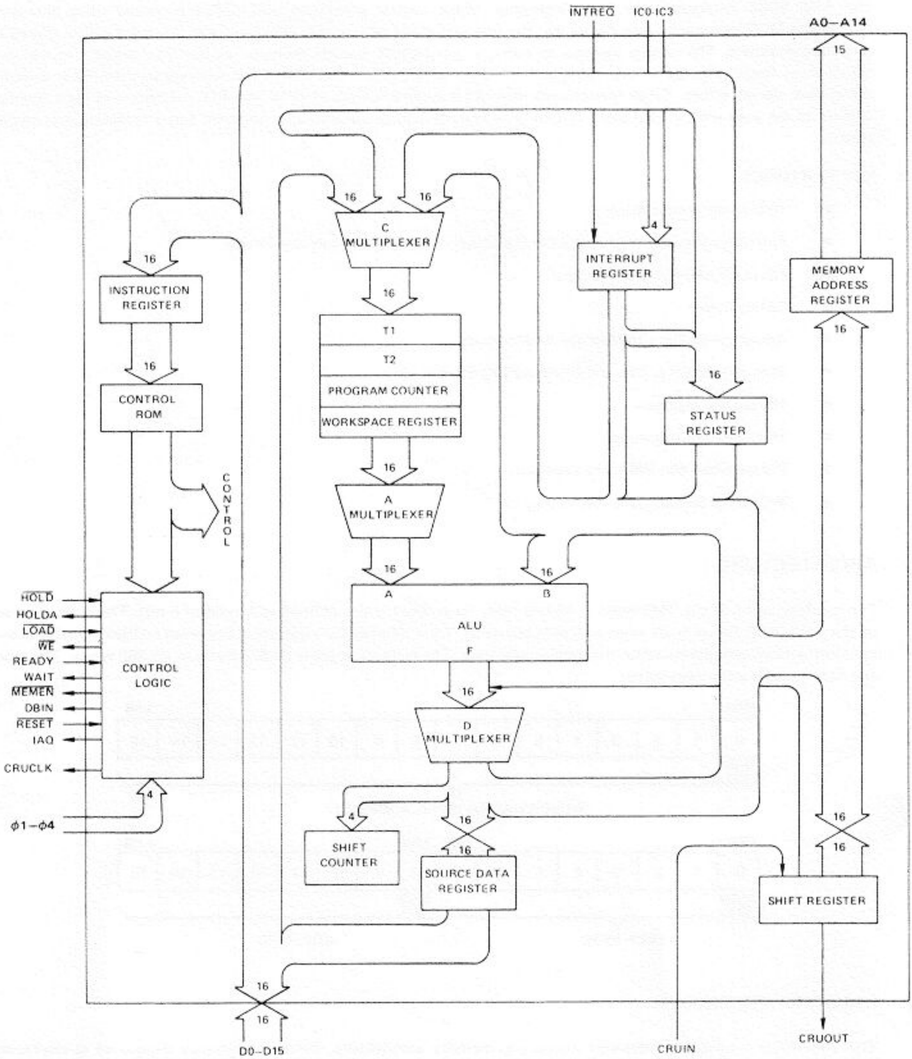


FIGURE 1 – ARCHITECTURE

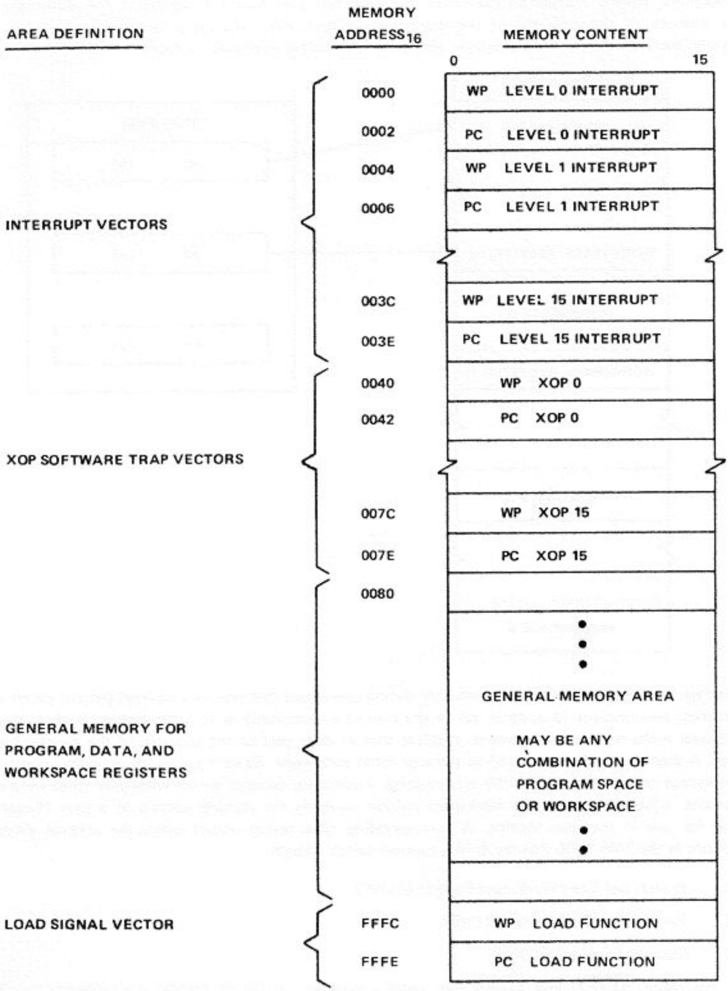
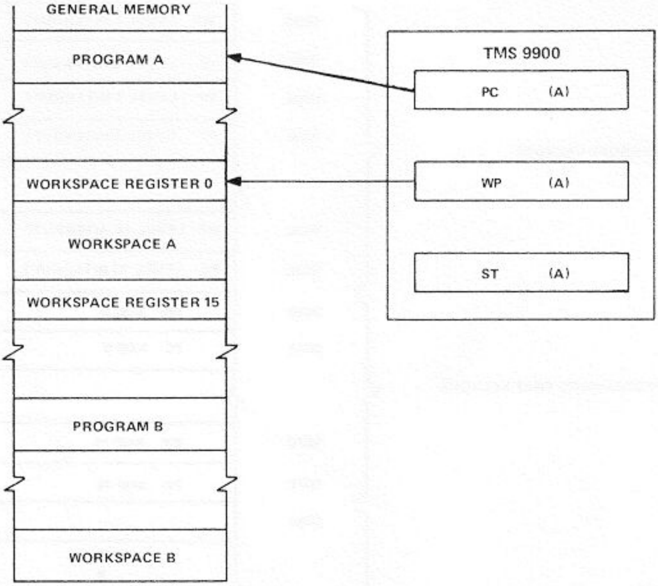


FIGURE 2 – MEMORY MAP

Three internal registers are accessible to the user. The program counter (PC) contains the address of the instruction following the current instruction being executed. This address is referenced by the processor to fetch the next instruction from memory and is then automatically incremented. The status register (ST) contains the present state of the processor and will be further defined in Section 3.4. The workspace pointer (WP) contains the address of the first word in the currently active set of workspace registers.

A workspace-register file occupies 16 contiguous memory words in the general memory area (see Figure 2). Each workspace register may hold data or addresses and function as operand registers, accumulators, address registers, or

index registers. During instruction execution, the processor addresses any register in the workspace by adding the register number to the contents of the workspace pointer and initiating a memory request for the word. The relationship between the workspace pointer and its corresponding workspace is shown below.



The workspace concept is particularly valuable during operations that require a context switch, which is a change from one program environment to another (as in the case of an interrupt) or to a subroutine. Such an operation, using a conventional multi-register arrangement, requires that at least part of the contents of the register file be stored and reloaded. A memory cycle is required to store or fetch each word. By exchanging the program counter, status register, and workspace pointer, the TMS 9900 accomplishes a complete context switch with only three store cycles and three fetch cycles. After the switch the workspace pointer contains the starting address of a new 16-word workspace in memory for use in the new routine. A corresponding time saving occurs when the original context is restored. Instructions in the TMS 9900 that result in a context switch include:

1. Branch and Load Workspace Pointer (BLWP)
2. Return from Subroutine (RTWP)
3. Extended Operation (XOP).

Device interrupts, $\overline{\text{RESET}}$, and $\overline{\text{LOAD}}$ also cause a context switch by forcing the processor to trap to a service subroutine.

2.2 INTERRUPTS

The TMS 9900 employs 16 interrupt levels with the highest priority level 0 and lowest level 15. Level 0 is reserved for the $\overline{\text{RESET}}$ function and all other levels may be used for external devices. The external levels may also be shared by several device interrupts, depending upon system requirements.

The TMS 9900 continuously compares the interrupt code (IC0 through IC3) with the interrupt mask contained in status-register bits 12 through 15. When the level of the pending interrupt is less than or equal to the enabling mask level (higher or equal priority interrupt), the processor recognizes the interrupt and initiates a context switch following

completion of the currently executing instruction. The processor fetches the new context WP and PC from the interrupt vector locations. Then, the previous context WP, PC, and ST are stored in workspace registers 13, 14, and 15, respectively, of the new workspace. The TMS 9900 then forces the interrupt mask to a value that is one less than the level of the interrupt being serviced, except for level-zero interrupt, which loads zero into the mask. This allows only interrupts of higher priority to interrupt a service routine. The processor also inhibits interrupts until the first instruction of the service routine has been executed to preserve program linkage should a higher priority interrupt occur. All interrupt requests should remain active until recognized by the processor in the device-service routine. The individual service routines must reset the interrupt requests before the routine is complete.

If a higher priority interrupt occurs, a second context switch occurs to service the higher priority interrupt. When that routine is complete, a return instruction (RTWP) restores the first service routine parameters to the processor to complete processing of the lower-priority interrupt. All interrupt subroutines should terminate with the return instruction to restore original program parameters. The interrupt-vector locations, device assignment, enabling-mask value, and the interrupt code are shown in Table 1.

TABLE 1
INTERRUPT LEVEL DATA

Interrupt Level	Vector Location (Memory Address In Hex)	Device Assignment	Interrupt Mask Values To Enable Respective Interrupts (ST12 thru ST15)	Interrupt Codes IC0 thru IC3
(Highest priority) 0	00	Reset	0 through F*	0000
1	04	External device	1 through F	0001
2	08	↓	2 through F	0010
3	0C		3 through F	0011
4	10		4 through F	0100
5	14		5 through F	0101
6	18		6 through F	0110
7	1C		7 through F	0111
8	20		8 through F	1000
9	24		9 through F	1001
10	28		A through F	1010
11	2C		B through F	1011
12	30		C through F	1100
13	34		D through F	1101
14	38		E and F	1110
(Lowest priority) 15	3C	External device	F only	1111

*Level 0 can not be disabled.

The TMS 9900 interrupt interface utilizes standard TTL components as shown in Figure 3. Note that for eight or less external interrupts a single SN74148 is required and for one external interrupt INTREQ is used as the interrupt signal with a hard-wired code IC0 through IC3.

2.3 INPUT/OUTPUT

The TMS 9900 utilizes a versatile direct command-driven I/O interface designated as the communications-register unit (CRU). The CRU provides up to 4096 directly addressable input bits and 4096 directly addressable output bits. Both input and output bits can be addressed individually or in fields of from 1 to 16 bits. The TMS 9900 employs three dedicated I/O pins (CRUIN, CRUOUT, and CRUCLK) and 12 bits (A3 through A14) of the address bus to interface with the CRU system. The processor instructions that drive the CRU interface can set, reset, or test any bit in the CRU array or move between memory and CRU data fields.

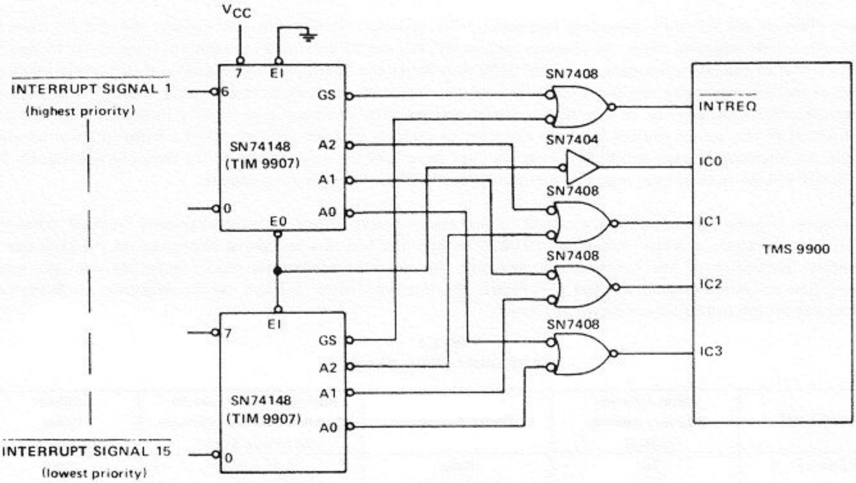


FIGURE 3 – TMS 9900 INTERRUPT INTERFACE

2.4 SINGLE-BIT CRU OPERATIONS

The TMS 9900 performs three single-bit CRU functions: test bit (TB), set bit to one (SBO), and set bit to zero (SBZ). To identify the bit to be operated upon, the TMS 9900 develops a CRU-bit address and places it on the address bus, A3 to A14.

For the two output operations (SBO and SBZ), the processor also generates a CRUCLK pulse, indicating an output operation to the CRU device, and places bit 7 of the instruction word on the CRUOUT line to accomplish the specified operation (bit 7 is a one for SBO and a zero for SBZ). A test-bit instruction transfers the addressed CRU bit from the CRUIN input line to bit 2 of the status register (EQUAL).

The TMS 9900 develops a CRU-bit address for the single-bit operations from the CRU-base address contained in workspace register 12 and the signed displacement count contained in bits 8 through 15 of the instruction. The displacement allows two's complement addressing from base minus 128 bits through base plus 127 bits. The base address from W12 is added to the signed displacement specified in the instruction and the result is loaded onto the address bus. Figure 4 illustrates the development of a single-bit CRU address.

2.5 MULTIPLE-BIT CRU OPERATIONS

The TMS 9900 performs two multiple-bit CRU operations: store communications register (STCR) and load communications register (LDCR). Both operations perform a data transfer from the CRU-to-memory or from memory-to-CRU as illustrated in Figure 5. Although the figure illustrates a full 16-bit transfer operation, any number of bits from 1 through 16 may be involved. The LDCR instruction fetches a word from memory and right-shifts it to serially transfer it to CRU output bits. If the LDCR involves eight or fewer bits, those bits come from the right-justified field within the addressed byte of the memory word. If the LDCR involves nine or more bits, those bits come from the right-justified field within the whole memory word. When transferred to the CRU interface, each successive bit receives an address that is sequentially greater than the address for the previous bit. This addressing mechanism results in an order reversal of the bits; that is, bit 15 of the memory word (or bit 7) becomes the lowest addressed bit in the CRU and bit 0 becomes the highest addressed bit in the CRU field.

An STCR instruction transfers data from the CRU to memory. If the operation involves a byte or less transfer, the transferred data will be stored right-justified in the memory byte with leading bits set to zero. If the operation involves from nine to 16 bits, the transferred data is stored right-justified in the memory word with leading bits set to zero.

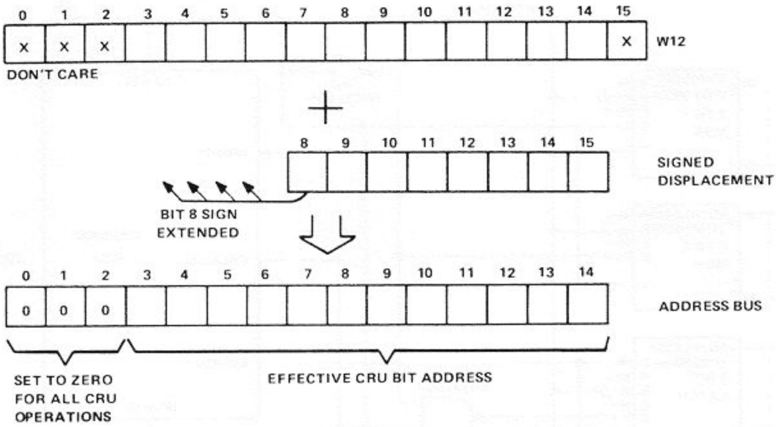


FIGURE 4 – TMS 9900 SINGLE-BIT CRU ADDRESS DEVELOPMENT

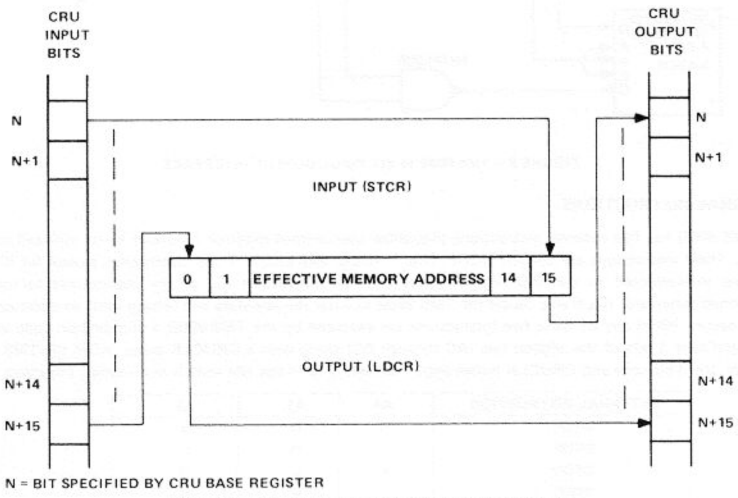


FIGURE 5 – TMS 9900 LDCR/STCR DATA TRANSFERS

When the input from the CRU device is complete, the first bit from the CRU is the least-significant-bit position in the memory word or byte.

Figure 6 illustrates how to implement a 16-bit input and a 16-bit output register in the CRU interface. CRU addresses are decoded as needed to implement up to 256 such 16-bit interface registers. In system application, however, only the exact number of interface bits needed to interface specific peripheral devices are implemented. It is not necessary to have a 16-bit interface register to interface an 8-bit device.

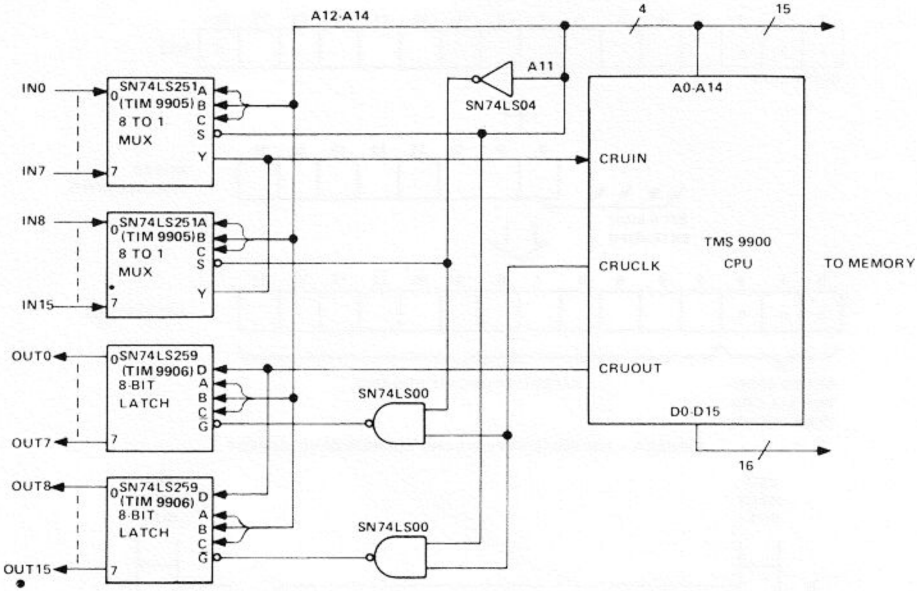


FIGURE 6 – TMS 9900 16-BIT INPUT/OUTPUT INTERFACE

2.6 EXTERNAL INSTRUCTIONS

The TMS 9900 has five external instructions that allow user-defined external functions to be initiated under program control. These instructions are CKON, CKOF, RSET, IDLE, and LREX. These mnemonics, except for IDLE, relate to functions implemented in the 990 minicomputer and do not restrict use of the instructions to initiate various user-defined functions. IDLE also causes the TMS 9900 to enter the idle state and remain until an interrupt, $\overline{\text{RESET}}$, or $\overline{\text{LOAD}}$ occurs. When any of these five instructions are executed by the TMS 9900, a unique 3-bit code appears on the most-significant 3 bits of the address bus (A0 through A2) along with a CRUCLK pulse. When the TMS 9900 is in an idle state, the 3-bit code and CRUCLK pulses occur repeatedly until the idle state is terminated. The codes are:

EXTERNAL INSTRUCTION	A0	A1	A2
LREX	H	H	H
CKOF	H	H	L
CKON	H	L	H
RSET	L	H	H
IDLE	L	H	L

Figure 7 illustrates typical external decode logic to implement these instructions. Note that a signal is generated to inhibit CRU decodes during external instructions.

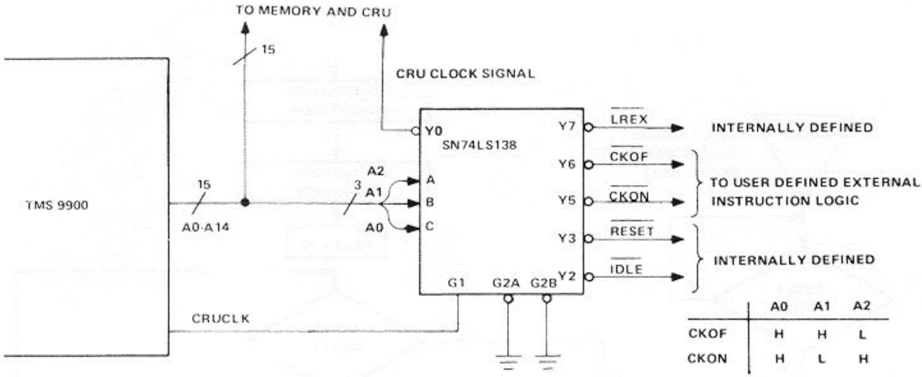


FIGURE 7 – EXTERNAL INSTRUCTION DECODE LOGIC

2.7 LOAD FUNCTION

The $\overline{LOAD}$ signal allows cold-start ROM loaders and front panels to be implemented for the TMS 9900. When active, $\overline{LOAD}$ causes the TMS 9900 to initiate an interrupt sequence immediately following the instruction being executed. Memory location FFFC is used to obtain the vector (WP and PC). The old PC, WP and ST are loaded into the new workspace and the interrupt mask is set to 0000. Then, program execution resumes using the new PC and WP.

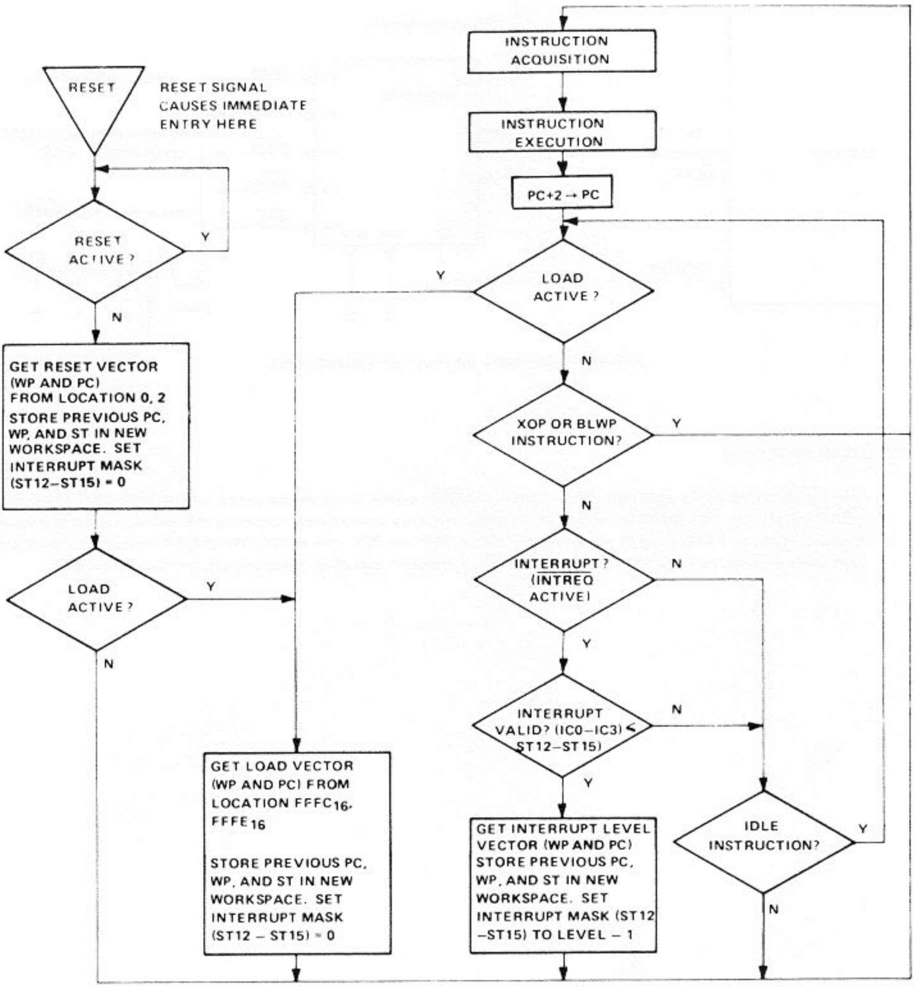


FIGURE 8 – TMS 9900 CPU FLOW CHART

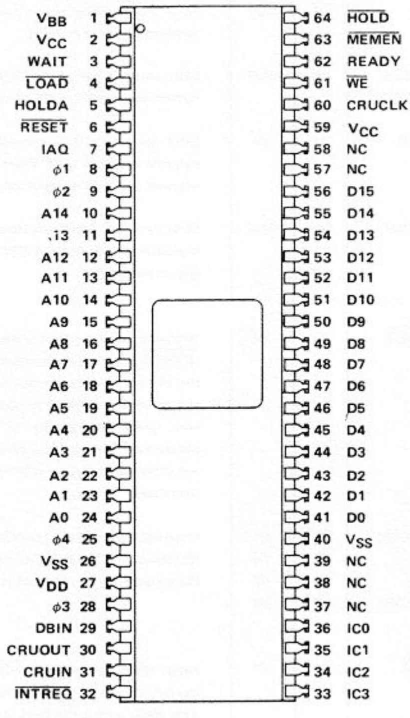
2.8 TMS 9900 PIN DESCRIPTION

Table 2 defines the TMS 9900 pin assignments and describes the function of each pin.

TABLE 2
TMS 9900 PIN ASSIGNMENTS AND FUNCTIONS

SIGNATURE	PIN	I/O	DESCRIPTION
			ADDRESS BUS
A0 (MSB)	24	OUT	A0 through A14 comprise the address bus. This 3-state bus provides the memory-address vector to the external-memory system when MEMEN is active and I/O-bit addresses and external-instruction addresses to the I/O system when MEMEN is inactive. The address bus assumes the high-impedance state when HOLDA is active.
A1	23	OUT	
A2	22	OUT	
A3	21	OUT	
A4	20	OUT	
A5	19	OUT	
A6	18	OUT	
A7	17	OUT	
A8	16	OUT	
A9	15	OUT	
A10	14	OUT	
A11	13	OUT	
A12	12	OUT	
A13	11	OUT	
A14 (LSB)	10	OUT	
			DATA BUS
D0 (MSB)	41	I/O	D0 through D15 comprise the bidirectional 3-state data bus. This bus transfers memory data to (when writing) and from (when reading) the external-memory system when MEMEN is active. The data bus assumes the high-impedance state when HOLDA is active.
D1	42	I/O	
D2	43	I/O	
D3	44	I/O	
D4	45	I/O	
D5	46	I/O	
D6	47	I/O	
D7	48	I/O	
D8	49	I/O	
D9	50	I/O	
D10	51	I/O	
D11	52	I/O	
D12	53	I/O	
D13	54	I/O	
D14	55	I/O	
D15 (LSB)	56	I/O	
			POWER SUPPLIES
V_{BB}	1		Supply voltage (−5 V NOM)
V_{CC}	2,59		Supply voltage (5 V NOM). Pins 2 and 59 must be connected in parallel.
V_{DD}	27		Supply voltage (12 V NOM)
V_{SS}	26,40		Ground reference. Pins 26 and 40 must be connected in parallel.
			CLOCKS
$\phi1$	8	IN	Phase-1 clock
$\phi2$	9	IN	Phase-2 clock
$\phi3$	28	IN	Phase-3 clock
$\phi4$	25	IN	Phase-4 clock

TMS 9900 PIN ASSIGNMENTS



NC – No internal connection

TABLE 2 (CONTINUED)

SIGNATURE	PIN	I/O	DESCRIPTION
			BUS CONTROL
DBIN	29	OUT	Data bus in. When active (high), DBIN indicates that the TMS 9900 has disabled its output buffers to allow the memory to place memory-read data on the data bus during $\overline{\text{MEMEN}}$. DBIN remains low in all other cases except when HOLDA is active.
$\overline{\text{MEMEN}}$	63	OUT	Memory enable. When active (low), $\overline{\text{MEMEN}}$ indicates that the address bus contains a memory address.
$\overline{\text{WE}}$	61	OUT	Write enable. When active (low), $\overline{\text{WE}}$ indicates that memory-write data is available from the TMS 9900 to be written into memory.
CRUCLK	60	OUT	CRU clock. When active (high), CRUCLK indicates that external interface logic should sample the output data on CRUOUT or should decode external instructions on A0 through A2.
CRUIN	31	IN	CRU data in. CRUIN, normally driven by 3-state or open-collector devices, receives input data from external interface logic. When the processor executes a STCR or TB instruction, it samples CRUIN for the level of the CRU input bit specified by the address bus (A3 through A14).
CRUOUT	30	OUT	CRU data out. Serial I/O data appears on the CRUOUT line when an LDCR, SBZ, or SBO instruction is executed. The data on CRUOUT should be sampled by external I/O interface logic when CRUCLK goes active (high).
			INTERRUPT CONTROL
$\overline{\text{INTREQ}}$	32	IN	Interrupt request. When active (low), $\overline{\text{INTREQ}}$ indicates that an external interrupt is requested. If $\overline{\text{INTREQ}}$ is active, the processor loads the data on the interrupt-code-input lines IC0 through IC3 into the internal interrupt-code-storage register. The code is compared to the interrupt mask bits of the status register. If equal or higher priority than the enabled interrupt level (interrupt code equal or less than status register bits 12 through 15) the TMS 9900 interrupt sequence is initiated. If the comparison fails, the processor ignores the request. $\overline{\text{INTREQ}}$ should remain active and the processor will continue to sample IC0 through IC3 until the program enables a sufficiently low priority to accept the request interrupt.
IC0 (MSB) IC1 IC2 IC3 (LSB)	36 35 34 33	IN IN IN IN	Interrupt codes. IC0 is the MSB of the interrupt code, which is sampled when $\overline{\text{INTREQ}}$ is active. When IC0 through IC3 are LLLH, the highest external-priority interrupt is being requested and when HHHH, the lowest-priority interrupt is being requested.
			MEMORY CONTROL
$\overline{\text{HOLD}}$	64	IN	Hold. When active (low), $\overline{\text{HOLD}}$ indicates to the processor that an external controller (e.g., DMA device) desires to utilize the address and data buses to transfer data to or from memory. The TMS 9900 enters the hold state following a hold signal when it has completed its present memory cycle.* The processor then places the address and data buses in the high-impedance state (along with $\overline{\text{WE}}$, $\overline{\text{MEMEN}}$, and DBIN) and responds with a hold-acknowledge signal (HOLDA). When $\overline{\text{HOLD}}$ is removed, the processor returns to normal operation.
HOLDA	5	OUT	Hold acknowledge. When active (high), HOLDA indicates that the processor is in the hold state and the address and data buses and memory control outputs ($\overline{\text{WE}}$, $\overline{\text{MEMEN}}$, and DBIN) are in the high-impedance state.
READY	62	IN	Ready. When active (high), READY indicates that memory will be ready to read or write during the next clock cycle. When not-ready is indicated during a memory operation, the TMS 9900 enters a wait state and suspends internal operation until the memory systems indicate ready.
WAIT	3	OUT	Wait. When active (high), WAIT indicates that the TMS 9900 has entered a wait state because of a not-ready condition from memory.

*If the cycle following the present memory cycle is also a memory cycle, it, too, is completed before the TMS9900 enters the hold state. The maximum number of consecutive memory cycles is three.

TABLE 2 (CONCLUDED)

SIGNATURE	PIN	I/O	DESCRIPTION
			TIMING AND CONTROL
IAQ	7	OUT	Instruction acquisition. IAQ is active (high) during any memory cycle when the TMS 9900 is acquiring an instruction. IAQ can be used to detect illegal op codes.
$\overline{\text{LOAD}}$	4	IN	Load. When active (low), $\overline{\text{LOAD}}$ causes the TMS 9900 to execute a nonmaskable interrupt with memory address $FFFC_{16}$ containing the trap vector (WP and PC). The load sequence begins after the instruction being executed is completed. $\overline{\text{LOAD}}$ will also terminate an idle state. If $\overline{\text{LOAD}}$ is active during the time $\overline{\text{RESET}}$ is released, then the $\overline{\text{LOAD}}$ trap will occur after the $\overline{\text{RESET}}$ function is completed. $\overline{\text{LOAD}}$ should remain active for one instruction period. IAQ can be used to determine instruction boundaries. This signal can be used to implement cold-start ROM loaders. Additionally, front-panel routines can be implemented using CRU bits as front-panel-interface signals and software-control routines to control the panel operations.
$\overline{\text{RESET}}$	6	IN	Reset. When active (low), $\overline{\text{RESET}}$ causes the processor to be reset and inhibits $\overline{\text{WE}}$ and CRUCLK. When $\overline{\text{RESET}}$ is released, the TMS 9900 then initiates a level-zero interrupt sequence that acquires WP and PC from locations 0000 and 0002, sets all status register bits to zero, and starts execution. $\overline{\text{RESET}}$ will also terminate an idle state. $\overline{\text{RESET}}$ must be held active for a minimum of three clock cycles.

2.9 TIMING

2.9.1 MEMORY

A basic memory read and write cycle is shown in Figure 9. The read cycle is shown with no wait states and the write cycle is shown with one wait state.

$\overline{\text{MEMEN}}$ goes active (low) during each memory cycle. At the same time that $\overline{\text{MEMEN}}$ is active, the memory address appears on the address bus bits A0 through A14. If the cycle is a memory-read cycle, DBIN will go active (high) at the same time $\overline{\text{MEMEN}}$ and A0 through A14 become valid. The memory-write signal $\overline{\text{WE}}$ will remain inactive (high) during a read cycle. If the read cycle is also an instruction acquisition cycle, IAQ will go active (high) during the cycle.

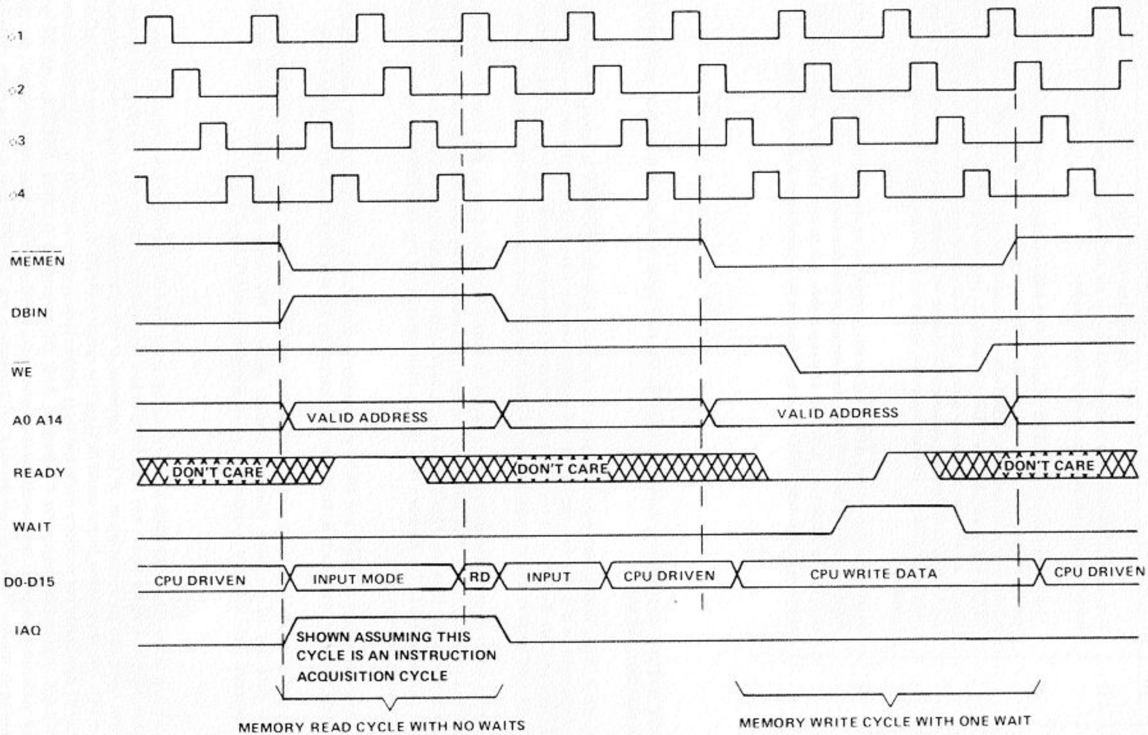
The READY signal, which allows extended memory cycles, is shown high during $\phi 1$ of the second clock cycle of the read operation. This indicates to the TMS 9900 that memory-read data will be valid during $\phi 1$ of the next clock cycle. If READY is low during $\phi 1$, then the TMS 9900 enters a wait state suspending internal operation until a READY is sensed during a subsequent $\phi 1$. The memory read data is then sampled by the TMS 9900 during the next $\phi 1$, which completes the memory-read cycle.

At the end of the read cycle, $\overline{\text{MEMEN}}$ and DBIN go inactive (high and low, respectively). The address bus may also change at this time, however, the data bus remains in the input mode for one clock cycle after the read cycle.

A write cycle is similar to the read cycle with the exception that $\overline{\text{WE}}$ goes active (low) as shown and valid write data appears on the data bus at the same time the address appears. The write cycle is shown as an example of a one-wait-state memory cycle. READY is low during $\phi 1$ resulting in the WAIT signal shown.

2.9.2 HOLD

Other interfaces may utilize the TMS 9900 memory bus by using the hold operation (illustrated in Figure 10) of the TMS 9900. When $\overline{\text{HOLD}}$ is active (low), the TMS 9900 enters the hold state at the next available non-memory cycle. Considering that there can be a maximum of two consecutive memory cycles, the maximum delay between $\overline{\text{HOLD}}$ going active to HOLDA going active (high) could be $t_{c(\phi)}$ (for setup) + (4+W) $t_{c(\phi)}$ + $t_{c(\phi)}$ (delay for HOLDA), where W is the number of wait states per memory cycle and $t_{c(\phi)}$ is the clock cycle time. When the TMS 9900 has entered the hold state, HOLDA goes active (high) and A0 through A15, D0 through D15 DBIN, $\overline{\text{MEMEN}}$, and $\overline{\text{WE}}$ go into a high-impedance state to allow other devices to use the memory buses. When $\overline{\text{HOLD}}$ goes inactive (high), the TMS 9900 resumes processing as shown. If hold occurs during a CRU operation, the TMS 9900 uses an extra clock cycle (after the removal of the $\overline{\text{HOLD}}$ signal) to reassert the CRU address providing the normal setup times for the CRU bit transfer that was interrupted.



RD = READ DATA

FIGURE 9 – TMS 9900 MEMORY BUS TIMING

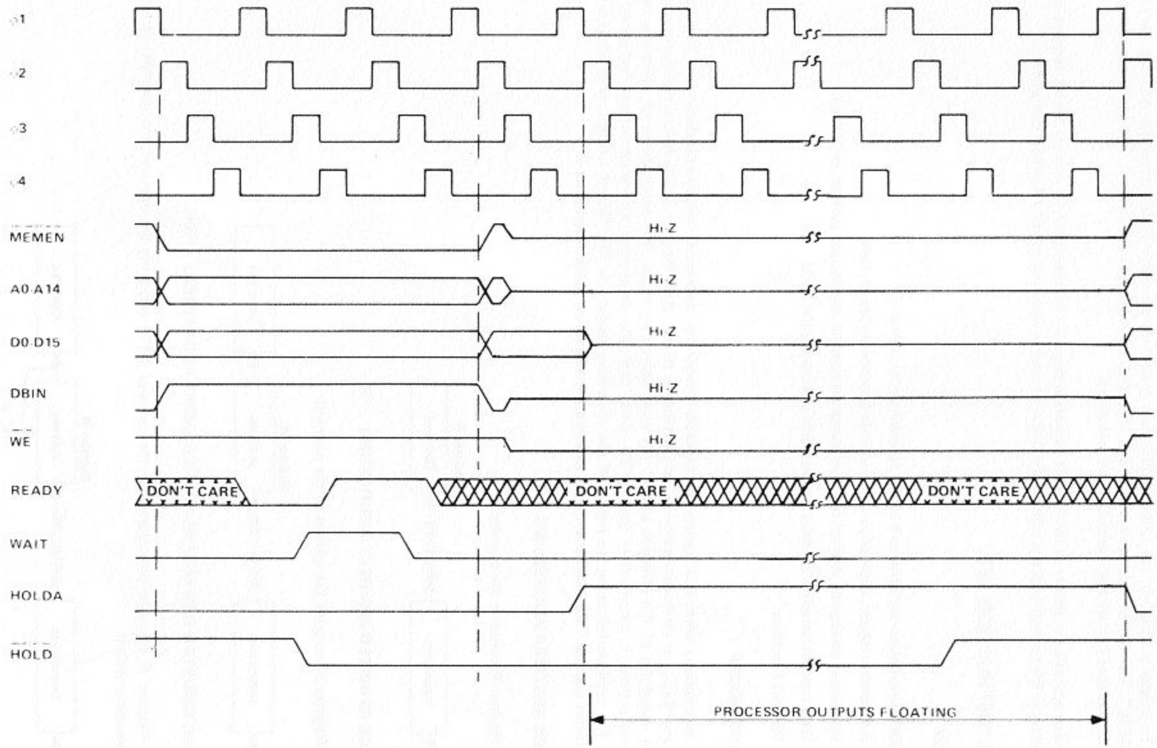


FIGURE 10 – TMS 9900 HOLD TIMING

2.9.3 CRU

CRU interface timing is shown in Figure 11. The timing for transferring two bits out and one bit in is shown. These transfers would occur during the execution of a CRU instruction. The other cycles of the instruction execution are not illustrated. To output a CRU bit, the CRU-bit address is placed on the address bus A0 through A14 and the actual bit data on CRUOUT. During the second clock cycle a CRU pulse is supplied by CRUCLK. This process is repeated until the number of bits specified by the instruction are completed.

The CRU input operation is similar in that the bit address appears on A0 through A14. During the subsequent cycle the TMS 9900 accepts the bit input data as shown. No CRUCLK pulses occur during a CRU input operation.

3. TMS 9900 INSTRUCTION SET

3.1 DEFINITION

Each TMS 9900 instruction performs one of the following operations:

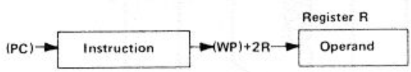
- Arithmetic, logical, comparison, or manipulation operations on data
- Loading or storage of internal registers (program counter, workspace pointer, or status)
- Data transfer between memory and external devices via the CRU
- Control functions.

3.2 ADDRESSING MODES

TMS 9900 instructions contain a variety of available modes for addressing random-memory data (e.g., program parameters and flags), or formatted memory data (character strings, data lists, etc.). The following figures graphically describe the derivation of the effective address for each addressing mode. The applicability of addressing modes to particular instructions is described in Section 3.5 along with the description of the operations performed by the instruction. The symbols following the names of the addressing modes [R, *R, *R+, @ LABEL, or @ TABLE (R)] are the general forms used by TMS 9900 assemblers to select the addressing mode for register R.

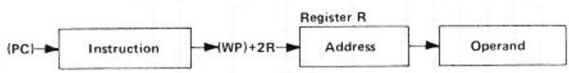
3.2.1 WORKSPACE REGISTER ADDRESSING R

Workspace Register R contains the operand.



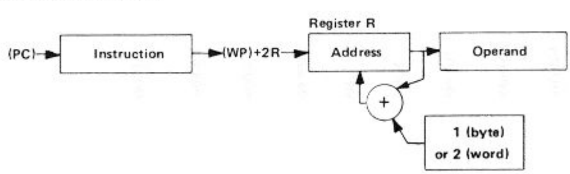
3.2.2 WORKSPACE REGISTER INDIRECT ADDRESSING *R

Workspace Register R contains the address of the operand.



3.2.3 WORKSPACE REGISTER INDIRECT AUTO INCREMENT ADDRESSING *R+

Workspace Register R contains the address of the operand. After acquiring the operand, the contents of workspace register R are incremented.



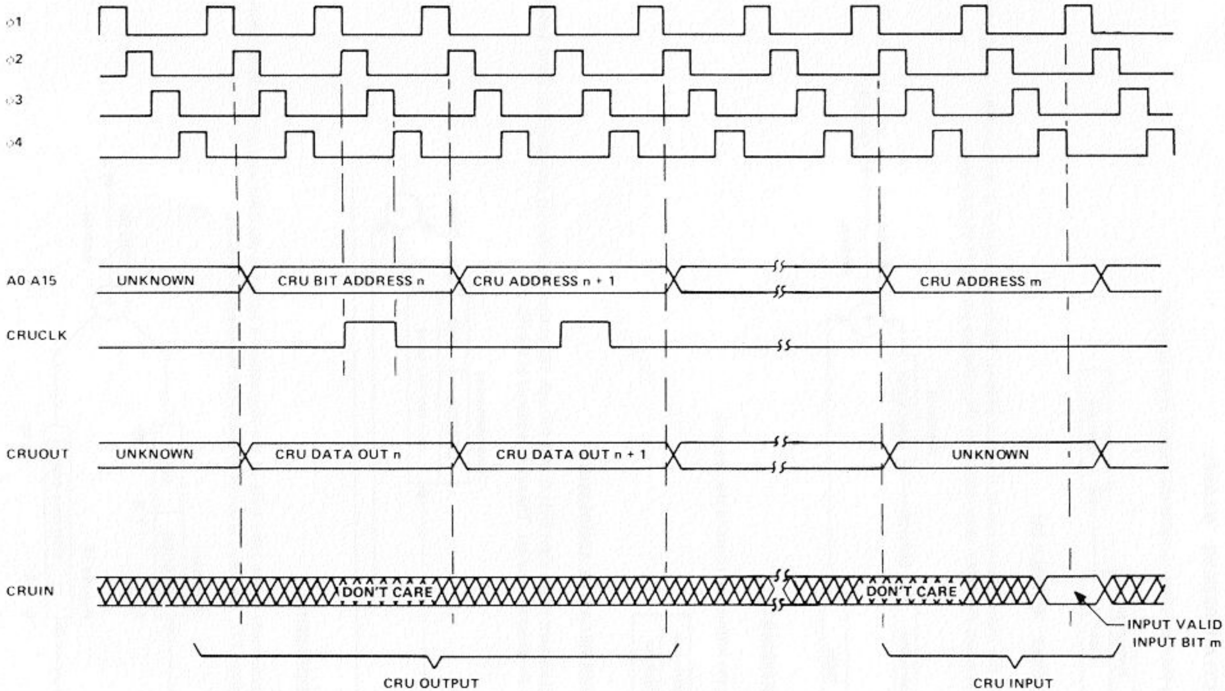
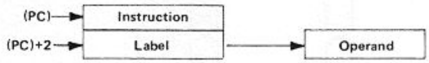


FIGURE 11 – TMS 9900 CRU INTERFACE TIMING

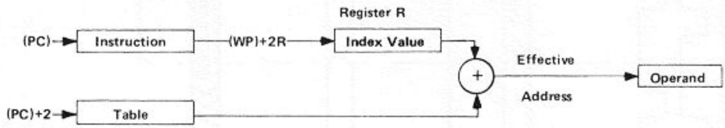
3.2.4 SYMBOLIC (DIRECT) ADDRESSING @ LABEL

The word following the instruction contains the address of the operand.



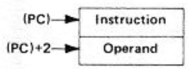
3.2.5 INDEXED ADDRESSING @ TABLE (R)

The word following the instruction contains the base address. Workspace register R contains the index value. The sum of the base address and the index value results in the effective address of the operand.



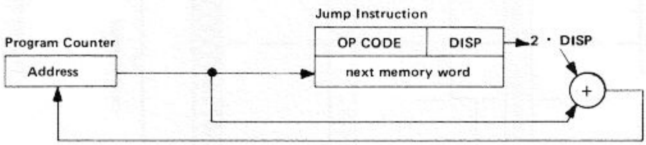
3.2.6 IMMEDIATE ADDRESSING

The word following the instruction contains the operand.



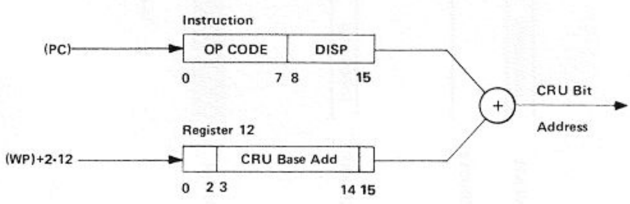
3.2.7 PROGRAM COUNTER RELATIVE ADDRESSING

The 8-bit signed displacement in the right byte (bits 8 through 15) of the instruction is multiplied by 2 and added to the updated contents of the program counter. The result is placed in the PC.



3.2.8 CRU RELATIVE ADDRESSING

The 8-bit signed displacement in the right byte of the instruction is added to the CRU base address (bits 3 through 14 of the workspace register 12). The result is the CRU address of the selected CRU bit.



3.3 TERMS AND DEFINITIONS

The following terms are used in describing the instructions of the TMS 9900:

TERM	DEFINITION
B	Byte indicator (1=byte, 0 = word)
C	Bit count
D	Destination address register
DA	Destination address
IOP	Immediate operand
LSB(n)	Least significant (right most) bit of (n)
MSB(n)	Most significant (left most) bit of (n)
N	Don't care
PC	Program counter
Result	Result of operation performed by instruction
S	Source address register
SA	Source address
ST	Status register
STn	Bit n of status register
T_D	Destination address modifier
T_S	Source address modifier
W	Workspace register
WRn	Workspace register n
(n)	Contents of n
a → b	a is transferred to b
|n|	Absolute value of n
+	Arithmetic addition
–	Arithmetic subtraction
AND	Logical AND
OR	Logical OR
⊕	Logical exclusive OR
$\overline{n}$	Logical complement of n

3.4 STATUS REGISTER

The status register contains the interrupt mask level and information pertaining to the instruction operation.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ST0	ST1	ST2	ST3	ST4	ST5	ST6	not used (=0)					ST12	ST13	ST14	ST15
L>	A>	=	C	O	P	X						Interrupt Mask			

BIT	NAME	INSTRUCTION	CONDITION TO SET BIT TO 1
ST0	LOGICAL GREATER THAN	C,CB	If MSB(SA) = 1 and MSB(DA) = 0, or if MSB(SA) = MSB(DA) and MSB of (DA) – (SA) = 1
		CI	If MSB(W) = 1 and MSB of IOP = 0, or if MSB(W) = MSB of IOP and MSB of IOP – (W) = 1
		ABS	If (SA) ≠ 0
		All Others	If result ≠ 0
ST1	ARITHMETIC GREATER THAN	C,CB	If MSB(SA) = 0 and MSB(DA) = 1, or if MSB(SA) = MSB(DA) and MSB(DA) – (SA) = 1
		CI	If MSB(W) = 0 and MSB of IOP = 1, or if MSB(W) = MSB of IOP and MSB of IOP – (W) = 1
		ABS	If MSB(SA) = 0 and (SA) ≠ 0
		All Others	If MSB of result = 0 and result ≠ 0

– Continued

BIT	NAME	INSTRUCTION	CONDITION TO SET BIT TO 1
ST2	EQUAL	C, CB	If (SA) = (DA)
		C1	If (W) = IOP
		COC	If (SA) and $(\overline{DA})$ = 0
		CZC	If (SA) and (DA) = 0
		TB	If CRUIN = 1
		ABS	If (SA) = 0
		All others	If result = 0
ST3	CARRY	A, AB, ABS, AI, DEC, DECT, INC, INCT, NEG, S, SB	If CARRY OUT = 1
		SLA, SRA, SRC, SRL	If last bit shifted out = 1
ST4	OVERFLOW	A, AB	If MSB(SA) = MSB(DA) and MSB of result ≠ MSB(DA)
		AI	If MSB(W) = MSB of IOP and MSB of result ≠ MSB(W)
		S, SB	If MSB(SA) ≠ MSB(DA) and MSB of result ≠ MSB(DA)
		DEC, DECT	If MSB(SA) = 1 and MSB of result = 0
		INC, INCT	If MSB(SA) = 0 and MSB of result = 1
		SLA	If MSB changes during shift
		DIV	If MSB(SA) = 0 and MSB(DA) = 1, or if MSB(SA) = MSB(DA) and MSB of (DA) − (SA) = 0
		ABS, NEG	If (SA) = 8000_{16}
ST5	PARITY	CB, MOVB	If (SA) has odd number of 1's
		LDCR, STCR	If 1 ≤ C ≤ 8 and (SA) has odd number of 1's
		AB, SB, SOCB, SZCB	If result has odd number of 1's
ST6	XOP	XOP	If XOP instruction is executed
ST12–ST15	INTERRUPT MASK	LIMI	If corresponding bit of IOP is 1
		RTWP	If corresponding bit of WR15 is 1

3.5 INSTRUCTIONS

3.5.1 Dual Operand Instructions with Multiple Addressing Modes for Source and Destination Operand

General format:

0 1 2	3	4 5	6 7 8 9	10 11	12 13 14 15
OP CODE	B	T_D	D	T_S	S

If B = 1 the operands are bytes and the operand addresses are byte addresses. If B = 0 the operands are words and the operand addresses are word addresses.

The addressing mode for each operand is determined by the T field of that operand.

T_S OR T_D	S OR D	ADDRESSING MODE	NOTES
00	0, 1, . . . 15	Workspace register	1
01	0, 1, . . . 15	Workspace register indirect	
10	0	Symbolic	4
10	1, 2, . . . 15	Indexed	2,4
11	0, 1, . . . 15	Workspace register indirect auto-increment	3

NOTES: 1. When a workspace register is the operand of a byte instruction (bit 3 = 1), the left byte (bits 0 through 7) is the operand and the right byte (bits 8 through 15) is unchanged.
2. Workspace register 0 may not be used for indexing.
3. The workspace register is incremented by 1 for byte instructions (bit 3 = 1) and is incremented by 2 for word instructions (bit 3 = 0).
4. When T_S = T_D = 10, two words are required in addition to the instruction word. The first word is the source operand base address and the second word is the destination operand base address.

MNEMONIC	OP CODE 0 1 2	B 3	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
A	1 0 1	0	Add	Yes	0-4	(SA)+(DA) → (DA)
AB	1 0 1	1	Add bytes	Yes	0-5	(SA)+(DA) → (DA)
C	1 0 0	0	Compare	No	0-2	Compare (SA) to (DA) and set appropriate status bits
CB	1 0 0	1	Compare bytes	No	0-2,5	Compare (SA) to (DA) and set appropriate status bits
S	0 1 1	0	Subtract	Yes	0-4	(DA) – (SA) → (DA)
SB	0 1 1	1	Subtract bytes	Yes	0-5	(DA) – (SA) → (DA)
SOC	1 1 1	0	Set ones corresponding	Yes	0-2	(DA) OR (SA) → (DA)
SOCB	1 1 1	1	Set ones corresponding bytes	Yes	0-2,5	(DA) OR (SA) → (DA)
SZC	0 1 0	0	Set zeroes corresponding	Yes	0-2	(DA) AND $(\overline{SA})$ → (DA)
SZCB	0 1 0	1	Set zeroes corresponding bytes	Yes	0-2,5	(DA) AND $(\overline{SA})$ → (DA)
MOV	1 1 0	0	Move	Yes	0-2	(SA) → (DA)
MOVB	1 1 0	1	Move bytes	Yes	0-2,5	(SA) → (DA)

3.5.2 Dual Operand Instructions with Multiple Addressing Modes for the Source Operand and Workspace Register Addressing for the Destination

General format:

0–5	6–9	10–11	12–15
OP CODE	D	T_S	S

The addressing mode for the source operand is determined by the T_S field.

T_S	S	ADDRESSING MODE	NOTES
00	0, 1, . . . 15	Workspace register	
01	0, 1, . . . 15	Workspace register indirect	
10	0	Symbolic	
10	1, 2, . . . 15	Indexed	1
11	0, 1, . . . 15	Workspace register indirect auto increment	2

NOTES: 1. Workspace register 0 may not be used for indexing.
2. The workspace register is incremented by 2.

MNEMONIC	OP CODE 0 1 2 3 4 5	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
COC	0 0 1 0 0 0	Compare ones corresponding	No	2	Test (D) to determine if 1's are in each bit position where 1's are in (SA). If so, set ST2.
CZC	0 0 1 0 0 1	Compare zeros corresponding	No	2	Test (D) to determine if 0's are in each bit position where 1's are in (SA). If so, set ST2.
XOR	0 0 1 0 1 0	Exclusive OR	Yes	0-2	(D) ⊕ (SA) → (D)
MPY	0 0 1 1 1 0	Multiply	No		Multiply unsigned (D) by unsigned (SA) and place unsigned 32-bit product in D (most significant) and D+1 (least significant). If WR15 is D, the next word in memory after WR15 will be used for the least significant half of the product.
DIV	0 0 1 1 1 1	Divide	No	4	If unsigned (SA) is less than or equal to unsigned (D), perform no operation and set ST4. Otherwise, divide unsigned (D) and (D+1) by unsigned (SA). Quotient → (D), remainder → (D+1). If D = 15, the next word in memory after WR 15 will be used for the remainder.

3.5.3 Extended Operation (XOP) Instruction

General format:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	1	0	1	1	D				T_S		S			

The T_S and S fields provide multiple mode addressing capability for the source operand. When the XOP is executed, ST6 is set and the following transfers occur:

$(40_{16} + 4D) \rightarrow (WP)$
$(42_{16} + 4D) \rightarrow (PC)$
SA → (new WR11)
(old WP) → (new WR13)
(old PC) → (new WR14)
(old ST) → (new WR15)

The TMS 9900 does not test interrupt requests ($\overline{INTREQ}$) upon completion of the XOP instruction.

3.5.4 Single Operand Instructions

General format:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OP CODE										T_S		S			

The T_S and S fields provide multiple mode addressing capability for the source operand.

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7 8 9	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
B	0 0 0 0 0 1 0 0 0 1	Branch	No	–	SA → (PC)
BL	0 0 0 0 0 1 1 0 1 0	Branch and link	No	–	(PC) → (WR11); SA → (PC)
BLWP	0 0 0 0 0 1 0 0 0 0	Branch and load workspace pointer	No	–	(SA) → (WP);(SA+1) → (PC); (old WP) → (new WR13); (old PC) → (new WR14); (old ST) → (new WR15); the interrupt input ($\overline{INTREQ}$) is not tested upon completion of the BLWP instruction.
CLR	0 0 0 0 0 1 0 0 1 1	Clear operand	No	–	0 → (SA)
SETO	0 0 0 0 0 1 1 1 0 0	Set to ones	No	–	$FFFF_{16}$ → (SA)
INV	0 0 0 0 0 1 0 1 0 1	Invert	Yes	0-2	$\overline{(SA)}$ → (SA)
NEG	0 0 0 0 0 1 0 1 0 0	Negate	Yes	0-4	–(SA) → (SA)
ABS	0 0 0 0 0 1 1 1 0 1	Absolute value*	No	0-4	|(SA)| → (SA)
SWPB	0 0 0 0 0 1 1 0 1 1	Swap bytes	No	–	(SA), bits 0 thru 7 → (SA), bits 8 thru 15; (SA), bits 8 thru 15 → (SA), bits 0 thru 7.
INC	0 0 0 0 0 1 0 1 1 0	Increment	Yes	0-4	(SA) + 1 → (SA)
INCT	0 0 0 0 0 1 0 1 1 1	Increment by two	Yes	0-4	(SA) + 2 → (SA)
DEC	0 0 0 0 0 1 1 0 0 0	Decrement	Yes	0-4	(SA) – 1 → (SA)
DECT	0 0 0 0 0 1 1 0 0 1	Decrement by two	Yes	0-4	(SA) – 2 → (SA)
X†	0 0 0 0 0 1 0 0 1 0	Execute	No	–	Execute the instruction at SA.

*Operand is compared to zero for status bit.
†If additional memory words for the execute instruction are required to define the operands of the instruction located at SA, these words will be accessed from PC and the PC will be updated accordingly. The instruction acquisition signal (IAQ) will not be true when the TMS 9900 accesses the instruction at SA. Status bits are affected in the normal manner for the instruction executed.

3.5.5 CRU Multiple-Bit Instructions

General format:

0–5	6–9	10–11	12–15
OP CODE	C	T_S	S

The C field specifies the number of bits to be transferred. If C = 0, 16 bits will be transferred. The CRU base register (WR12, bits 3 through 14) defines the starting CRU bit address. The bits are transferred serially and the CRU address is incremented with each bit transfer, although the contents of WR12 is not affected. T_S and S provide multiple mode addressing capability for the source operand. If 8 or fewer bits are transferred (C = 1 through 8), the source address is a byte address. If 9 or more bits are transferred (C = 0, 9 through 15), the source address is a word address. If the source is addressed in the workspace register indirect auto increment mode, the workspace register is incremented by 1 if C = 1 through 8, and is incremented by 2 otherwise.

MNEMONIC	OP CODE 0 1 2 3 4 5	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
LDCR	0 0 1 1 0 0	Load communcation register	Yes	0-2,5†	Beginning with LSB of (SA), transfer the specified number of bits from (SA) to the CRU.
STCR	0 0 1 1 0 1	Store communcation register	Yes	0-2,5†	Beginning with LSB of (SA), transfer the specified number of bits from the CRU to (SA). Load unfilled bit positions with 0.

†ST5 is affected only if $1 \leq C \leq 8$.

3.5.6 CRU Single-Bit Instructions

General format:

0–7	8–15
OP CODE	SIGNED DISPLACEMENT

CRU relative addressing is used to address the selected CRU bit.

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7	MEANING	STATUS BITS AFFECTED	DESCRIPTION
SBO	0 0 0 1 1 1 0 1	Set bit to one	–	Set the selected CRU output bit to 1.
SBZ	0 0 0 1 1 1 1 0	Set bit to zero	–	Set the selected CRU output bit to 0.
TB	0 0 0 1 1 1 1 1	Test bit	2	If the selected CRU input bit = 1, set ST2.

3.5.7 Jump Instructions

General format:

0–7	8–15
OP CODE	DISPLACEMENT

Jump instructions cause the PC to be loaded with the value selected by PC relative addressing if the bits of ST are at specified values. Otherwise, no operation occurs and the next instruction is executed since PC points to the next instruction. The displacement field is a word count to be added to PC. Thus, the jump instruction has a range of −128 to 127 words from memory-word address following the jump instruction. No ST bits are affected by jump instruction.

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7	MEANING	ST CONDITION TO LOAD PC
JEQ	0 0 0 1 0 0 1 1	Jump equal	ST2 = 1
JGT	0 0 0 1 0 1 0 1	Jump greater than	ST1 = 1
JH	0 0 0 1 1 0 1 1	Jump high	ST0 = 1 and ST2 = 0
JHE	0 0 0 1 0 1 0 0	Jump high or equal	ST0 = 1 or ST2 = 1
JL	0 0 0 1 1 0 1 0	Jump low	ST0 = 0 and ST2 = 0
JLE	0 0 0 1 0 0 1 0	Jump low or equal	ST0 = 0 or ST2 = 1
JLT	0 0 0 1 0 0 0 1	Jump less than	ST1 = 0 and ST2 = 0
JMP	0 0 0 1 0 0 0 0	Jump unconditional	unconditional
JNC	0 0 0 1 0 1 1 1	Jump no carry	ST3 = 0
JNE	0 0 0 1 0 1 1 0	Jump not equal	ST2 = 0
JNO	0 0 0 1 1 0 0 1	Jump no overflow	ST4 = 0
JOC	0 0 0 1 1 0 0 0	Jump on carry	ST3 = 1
JOP	0 0 0 1 1 1 0 0	Jump odd parity	ST5 = 1

3.5.8 Shift Instructions

General format:

0–7	8–11	12–15
OP CODE	C	W

If C = 0, bits 12 through 15 of WR0 contain the shift count. If C = 0 and bits 12 through 15 of WR0 = 0, the shift count is 16.

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
SLA	0 0 0 0 1 0 1 0	Shift left arithmetic	Yes	0-4	Shift (W) left. Fill vacated bit positions with 0.
SRA	0 0 0 0 1 0 0 0	Shift right arithmetic	Yes	0-3	Shift (W) right. Fill vacated bit positions with original MSB of (W).
SRC	0 0 0 0 1 0 1 1	Shift right circular	Yes	0-3	Shift (W) right. Shift previous LSB into MSB.
SRL	0 0 0 0 1 0 0 1	Shift right logical	Yes	0-3	Shift (W) right. Fill vacated bit positions with 0's.

3.5.9 Immediate Register Instructions

General format:

0–10	11	12–15
OP CODE	N	W
IOP		

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7 8 9 10	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
AI	0 0 0 0 0 0 1 0 0 0 1	Add immediate	Yes	0-4	(W) + IOP → (W)
ANDI	0 0 0 0 0 0 1 0 0 1 0	AND immediate	Yes	0-2	(W) AND IOP → (W)
CI	0 0 0 0 0 0 1 0 1 0 0	Compare immediate	Yes	0-2	Compare (W) to IOP and set appropriate status bits
LI	0 0 0 0 0 0 1 0 0 0 0	Load immediate	Yes	0-2	IOP → (W)
ORI	0 0 0 0 0 0 1 0 0 1 1	OR immediate	Yes	0-2	(W) OR IOP → (W)

3.5.10 Internal Register Load Immediate Instructions

General format:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OP CODE											N				
IOP															

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7 8 9 10	MEANING	DESCRIPTION
LWPI	0 0 0 0 0 0 1 0 1 1 1	Load workspace pointer immediate	IOP → (WP), no ST bits affected
LIMI	0 0 0 0 0 0 1 1 0 0 0	Load interrupt mask	IOP, bits 12 thru 15 → ST12 thru ST15

3.5.11 Internal Register Store Instructions

General format:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OP CODE											N	W			

No ST bits are affected.

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7 8 9 10	MEANING	DESCRIPTION
STST	0 0 0 0 0 0 1 0 1 1 0	Store status register	(ST) → (W)
STWP	0 0 0 0 0 0 1 0 1 0 1	Store workspace pointer	(WP) → (W)

3.5.12 Return Workspace Pointer (RTWP) Instruction

General format:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	1	1	1	0	0	N				

The RTWP instruction causes the following transfers to occur:

(WR15) → (ST)
(WR14) → (PC)
(WR13) → (WP)

3.5.13 External Instructions

General format:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OP CODE											N				

External instructions cause the three most-significant address lines (A0 through A2) to be set to the below-described levels and the CRUCLK line to be pulsed, allowing external control functions to be initiated.

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7 8 9 10	MEANING	STATUS BITS AFFECTED	DESCRIPTION	ADDRESS BUS A0	A1	A2
IDLE	0 0 0 0 0 0 1 1 0 1 0	Idle	—	Suspend TMS 9900 instruction execution until an interrupt, LOAD, or RESET occurs	L	H	L
RSET	0 0 0 0 0 0 1 1 0 1 1	Reset	12–15	0 → ST12 thru ST15	L	H	H
CKOF	0 0 0 0 0 0 1 1 1 1 0	User defined		———	H	H	L
CKON	0 0 0 0 0 0 1 1 1 0 1	User defined		———	H	L	H
LREX	0 0 0 0 0 0 1 1 1 1 1	User defined		———	H	H	H

3.6 TMS 9900 INSTRUCTION EXECUTION TIMES

Instruction execution times for the TMS 9900 are a function of:

1) Clock cycle time, $t_{c(\phi)}$
2) Addressing mode used where operands have multiple addressing mode capability
3) Number of wait states required per memory access.

Table 3 lists the number of clock cycles and memory accesses required to execute each TMS 9900 instruction. For instructions with multiple addressing modes for either or both operands, the table lists the number of clock cycles and memory accesses with all operands addressed in the workspace-register mode. To determine the additional number of clock cycles and memory accesses required for modified addressing, add the appropriate values from the referenced tables. The total instruction-execution time for an instruction is:

$$T = t_{c(\phi)} \quad (C + W \cdot M)$$

where:

T = total instruction execution time;
$t_{c(\phi)}$ = clock cycle time;
C = number of clock cycles for instruction execution plus address modification;
W = number of required wait states per memory access for instruction execution plus address modification;
M = number of memory accesses.

TABLE 3
INSTRUCTION EXECUTION TIMES

INSTRUCTION	CLOCK CYCLES C	MEMORY ACCESS M	ADDRESS MODIFICATION†	
			SOURCE	DEST
A	14	4	A	A
AB	14	4	B	B
ABS (MSB = 0)	12	2	A	–
(MSB = 1)	14	3	A	–
AI	14	4	–	–
ANDI	14	4	–	–
B	8	2	A	–
BL	12	3	A	–
BLWP	26	6	A	–
C	14	3	A	A
CB	14	3	B	B
CI	14	3	–	–
CKOF	12	1	–	–
CKON	12	1	–	–
CLR	10	3	A	–
COC	14	3	A	–
CZC	14	3	A	–
DEC	10	3	A	–
DECT	10	3	A	–
DIV (ST4 is set)	16	3	A	–
DIV (ST4 is reset)*	92-124	6	A	–
IDLE	12	1	–	–
INC	10	3	A	–
INCT	10	3	A	–
INV	10	3	A	–
Jump (PC is changed)	10	1	–	–
(PC is not changed)	8	1	–	–
LDCR (C = 0)	52	3	A	–
(1 ≤ C ≤ 8)	20+2C	3	B	–
(9 ≤ C ≤ 15)	20+2C	3	A	–
LI	12	3	–	–
LIMI	16	2	–	–
LREX	12	1	–	–
$\overline{\text{RESET}}$ function	26	5	–	–
$\overline{\text{LOAD}}$ function	22	5	–	–
Interrupt context switch	22	5	–	–

INSTRUCTION	CLOCK CYCLES C	MEMORY ACCESS M	ADDRESS MODIFICATION†	
			SOURCE	DEST
LWPI	10	2	–	–
MOV	14	4	A	A
MOVB	14	4	B	B
MPY	52	5	A	–
NEG	12	3	A	–
ORI	14	4	–	–
RSET	12	1	–	–
RTWP	14	4	–	–
S	14	4	A	A
SB	14	4	B	B
SBO	12	2	–	–
SBZ	12	2	–	–
SETO	10	3	A	–
Shift (C ≠ 0)	12+2C	3	–	–
(C=0, Bits 12–15 of WR0=0)	52	4	–	–
(C=0, Bits 12–15 of WR0=N≠0)	20+2N	4	–	–
SOC	14	4	A	A
SOCB	14	4	B	B
STCR (C=0)	60	4	A	–
(1 ≤ C ≤ 7)	42	4	B	–
(C=8)	44	4	B	–
(9 ≤ C ≤ 15)	58	4	A	–
STST	8	2	–	–
STWP	8	2	–	–
SWPB	10	3	A	–
SZC	14	4	A	A
SZCB	14	4	B	B
TB	12	2	–	–
X **	8	2	A	–
XOP	36	8	A	–
XOR	14	4	A	–
Undefined op codes: 0000-01FF, 0320-033F, 0C00-0FFF, 0780-07FF	6	1	–	–

*Execution time is dependent upon the partial quotient after each clock cycle during execution.
**Execution time is added to the execution time of the instruction located at the source address minus 4 clock cycles and 1 memory access time.
†The letters A and B refer to the respective tables that follow.

ADDRESS MODIFICATION – TABLE A

ADDRESSING MODE	CLOCK CYCLES C	MEMORY ACCESSES M
WR (T_S or T_D = 00)	0	0
WR indirect (T_S or T_D = 01)	4	1
WR indirect auto-increment (T_S or T_D = 11)	8	2
Symbolic (T_S or T_D = 10, S or D = 0)	8	1
Indexed (T_S or T_D = 10, S or D ≠ 0)	8	2

ADDRESS MODIFICATION – TABLE B

ADDRESSING MODE	CLOCK CYCLES C	MEMORY ACCESSES M
WR (T_S or T_D = 00)	0	0
WR indirect (T_S or T_D = 01)	4	1
WR indirect auto-increment (T_S or T_D = 11)	6	2
Symbolic (T_S or T_D = 10, S or D = 0)	8	1
Indexed (T_S or T_D = 10, S or D ≠ 0)	8	2

As an example, the instruction MOVB is used in a system with $t_{c(\phi)}$ = 0.333 μs and no wait states are required to access memory. Both operands are addressed in the workspace register mode:

$$T = t_{c(\phi)}\ (C + W \cdot M) = 0.333\ (14 + 0 \cdot 4)\ \mu s = 4.662\ \mu s.$$

If two wait states per memory access were required, the execution time is:

$$T = 0.333\ (14 + 2 \cdot 4)\ \mu s = 7.326\ \mu s.$$

If the source operand was addressed in the symbolic mode and two wait states were required:

$$T = t_{c(\phi)}\ (C + W \cdot M)$$
$$C = 14 + 8 = 22$$
$$M = 4 + 1 = 5$$
$$T = 0.333\ (22 + 2 \cdot 5)\ \mu s = 10.656\ \mu s.$$

4. TMS 9900 ELECTRICAL AND MECHANICAL SPECIFICATIONS

4.1 ABSOLUTE MAXIMUM RATINGS OVER OPERATING FREE-AIR TEMPERATURE RANGE (UNLESS OTHERWISE NOTED)*

Supply voltage, V_{CC} (see Note 1) . . . . −0.3 to 20 V
Supply voltage, V_{DD} (see Note 1) . . . . −0.3 to 20 V
Supply voltage, V_{SS} (see Note 1) . . . . −0.3 to 20 V
All input voltages (see Note 1) . . . . −0.3 to 20 V
Output voltage (with respect to V_{SS}) . . . . −2 V to 7 V
Continuous power dissipation . . . . 1.2 W
Operating free-air temperature range . . . . 0°C to 70°C
Storage temperature range . . . . −55°C to 150°C

*Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 1: Under absolute maximum ratings voltage values are with respect to the most negative supply, V_{BB} (substrate), unless otherwise noted. Throughout the remainder of this section, voltage values are with respect to V_{SS}.

4.2 RECOMMENDED OPERATING CONDITIONS

	MIN	NOM	MAX	UNIT
Supply voltage, V_{BB}	−5.25	−5	−4.75	V
Supply voltage, V_{CC}	4.75	5	5.25	V
Supply voltage, V_{DD}	11.4	12	12.6	V
Supply voltage, V_{SS}		0		V
High-level input voltage, V_{IH} (all inputs except clocks)	2.2	2.4	$V_{CC}+1$	V
High-level clock input voltage, $V_{IH(\phi)}$	$V_{DD}-2$		V_{DD}	V
Low-level input voltage, V_{IL} (all inputs except clocks)	−1	0.4	0.8	V
Low-level clock input voltage, $V_{IL(\phi)}$	−0.3	0.3	0.6	V
Operating free-air temperature, T_A	0		70	°C

4.3 ELECTRICAL CHARACTERISTICS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS (UNLESS OTHERWISE NOTED)

PARAMETER			TEST CONDITIONS	MIN	TYP†	MAX	UNIT
I_I	Input current	Data bus during DBIN	$V_I = V_{SS}$ to V_{CC}		±50	±100	μA
		WE, MEMEN, DBIN, Address bus, Data bus during HOLDA	$V_I = V_{SS}$ to V_{CC}		±50	±100	μA
		Clock	$V_I = -0.3$ to 12.6 V		±25	±75	μA
		Any other inputs	$V_I = V_{SS}$ to V_{CC}		±1	±10	μA
V_{OH}	High-level output voltage		$I_O = -0.4$ mA	2.4		V_{CC}	V
V_{OL}	Low-level output voltage		$I_O = 3.2$ mA			0.65	V
			$I_O = 2$ mA			0.50	V
I_{BB}	Supply current from V_{BB}				0.1	1	mA
I_{CC}	Supply current from V_{CC}				50	75	mA
I_{DD}	Supply current from V_{DD}				25	45	mA
C_i	Input capacitance (any inputs except clock and data bus)		f = 1 MHz, V_{BB} = -5 V, unmeasured pins at V_{SS}		10	15	pF
$C_{i(\phi1)}$	Clock-1 input capacitance		f = 1 MHz, V_{BB} = -5 V, unmeasured pins at V_{SS}		100	150	pF
$C_{i(\phi2)}$	Clock-2 input capacitance		f = 1 MHz, V_{BB} = -5 V, unmeasured pins at V_{SS}		150	200	pF
$C_{i(\phi3)}$	Clock-3 input capacitance		f = 1 MHz, V_{BB} = -5 V, unmeasured pins at V_{SS}		100	150	pF
$C_{i(\phi4)}$	Clock-4 input capacitance		f = 1 MHz, V_{BB} = -5V, unmeasured pins at V_{SS}		100	150	pF
C_{DB}	Data bus capacitance		f = 1 MHz, V_{BB} = -5 V, unmeasured pins at V_{SS}		15	25	pF
C_o	Output capacitance (any output except data bus)		f = 1 MHz, V_{BB} = -5 V, unmeasured pins at V_{SS}		10	15	pF

† All typical values are at T_A = 25°C and nominal voltages.

4.4 TIMING REQUIREMENTS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS

PARAMETER		MIN	NOM	MAX	UNIT
$t_{c(\phi)}$	Clock cycle time	300	333	500	ns
$t_{r(\phi)}$	Clock rise time	5	12		ns
$t_{f(\phi)}$	Clock fall time	10	12		ns
$t_{w(\phi)}$	Clock pulse width, high level	40	45	100	ns
$t_{s(\phi)}$	Clock spacing, time between any two adjacent clock pulses	0	5		ns
$t_{d(\phi)}$	Time between rising edge valid any two adjacent clock pulses	73	83		ns
t_{su}	Data or control setup time before clock 1	30			ns
t_h	Data hold time after clock 1	10			ns

4.5 SWITCHING CHARACTERISTICS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
$t_{PLH(B)}$ or $t_{PHL(B)}$	All other outputs	C_L = 200 pF		20	40	ns
$t_{PLH(C)}$ or $t_{PHL(C)}$	Propagation delay CRUCLK, WE, MEMEN, WAIT, DBIN				30	ns

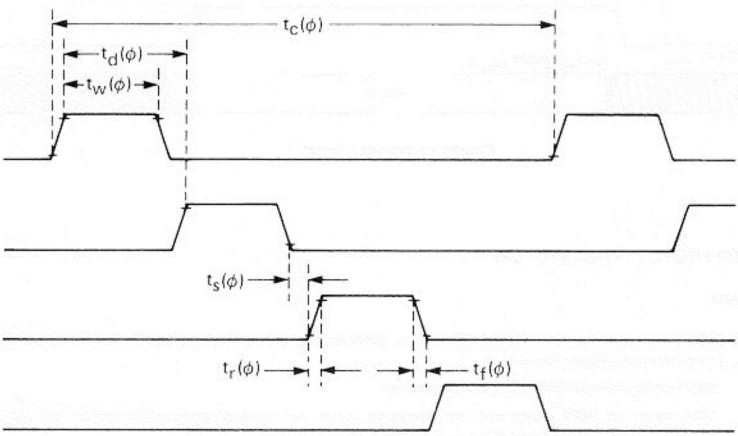


FIGURE 12 – CLOCK TIMING

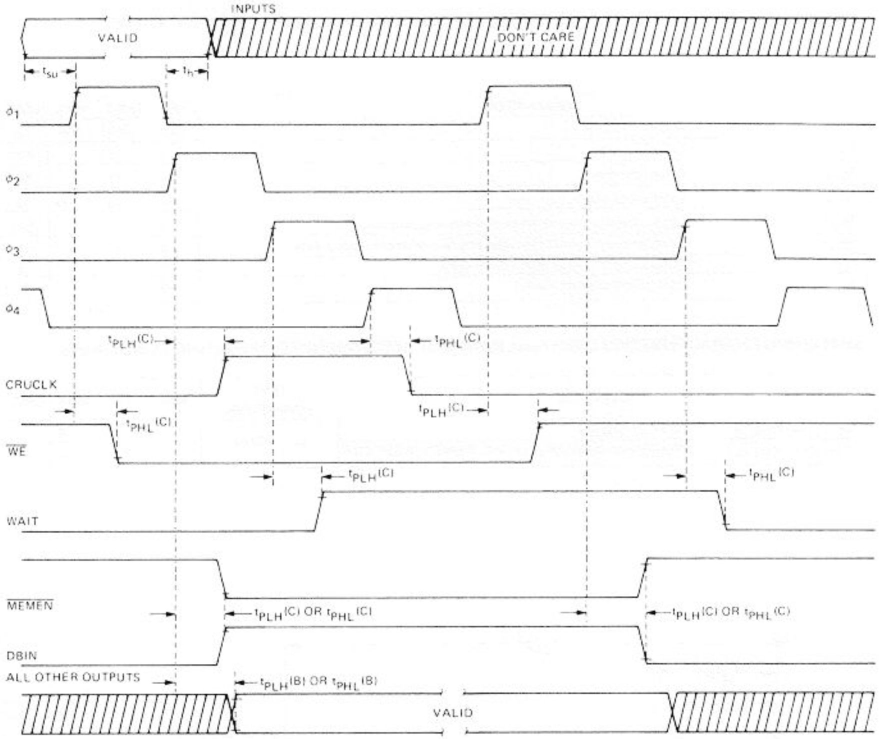


FIGURE 13–SIGNAL TIMING

5. TMS 9900 PROTOTYPING SYSTEM

5.1 HARDWARE

The TMS 9900 prototyping system enables the user to generate and debug software and to debug I/O controller interfaces. The prototyping system consists of:

- 990/4 computer with TMS 9900 microprocessor
- 1024 bytes of ROM containing the bootstrap loader for loading prototyping system software, the front-panel and maintenance utility, and the CPU self-testing feature
- 16,896 bytes of RAM with provisions for expansion up to 57,334 bytes of RAM
- Programmable-write-protect feature for RAM
- Interface for Texas Instruments Model 733 ASR* Electronic Data Terminal with provisions for up to five additional interface moculdes

*Requires remote device control and 1200 baud EIA interface option on 733 ASR.

- Available with Texas Instruments Model 733 ASR Electronic Data Terminal
- 7-inch-high table-top chassis
- Programmer's front panel with controls for run, halt, single-instruction execute, and entering and displaying memory or register contents
- Power supply with the following voltages:
  5 V dc @ 20 A
  12 V dc @ 2 A
  −12 V dc @ 1 A
  −5 V dc @ 0.1 A
- Complete hardware and software documentation.

5.2 SYSTEM CONSOLE

The system console for the prototyping system is the 733 ASR, which provides keyboard entry, 30-character-per-second thermal printer, and dual cassette drives for program loading and storage.

5.3 SOFTWARE

The following software is provided on cassette for loading into the prototyping system:

- Debug Monitor – Provides full control of the prototyping system during program development and includes single instruction, multiple breakpoints, and entry and display capability for register and memory contents for debugging user software under 733 ASR console control.
- One-Pass Assembler – Converts source code stored on cassette to relocatable object on cassette and generates program listing. (Object is upward compatible with other 990 series assemblers).
- Linking Loader – Allows loading of absolute and relocatable object modules and links object modules as they are loaded.
- Source Editor – Enables user modification of both source and object from cassette with resultant storage on cassette.
- Trace Routine – Allows user to monitor status of computer at completion of each instruction.
- PROM Programming/Documentation Facility – Provides documentation for ROM mask generation, or communicates directly with the optional PROM Programmer Unit.

5.4 OPTIONS

The following optional equipment is offered for the prototyping system:

- Battery-pack/standby-power supply
- PROM programming unit and adapter boards
- Universal wire-wrap modules
- Expansion RAM modules
- Expansion EPROM modules
- I/O modules and other interfaces
- Rack-mounted version
- International ac voltage option

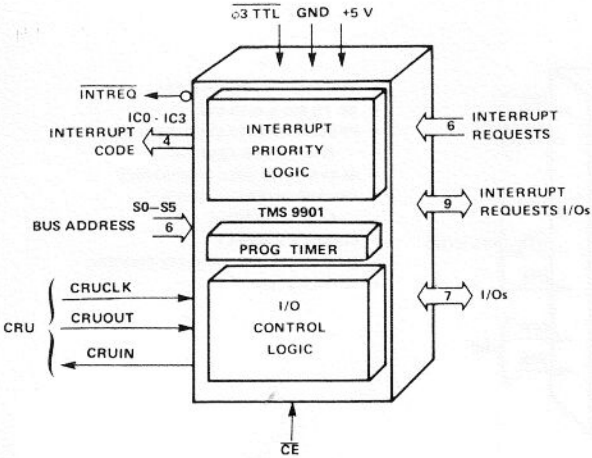
6. TMS 9900 SUPPORT CIRCUITS

MEMORY DEVICE	ORGANIZATION/FUNCTION	I/O STRUCTURE	PACKAGE	ACCESS TIME
		RAMS		
TMS 4036-2	64 x 8 static	Common bus	20 pin	450 ns MAX
TMS 4033	1024 x 1 static	Dedicated bus	16 pin	450 ns MAX
TMS 4039-2	256 x 4 static	Dedicated bus	22 pin	450 ns MAX
TMS 4042-2	256 x 4 static	Common bus	18 pin	450 ns MAX
TMS 4043-2	256 x 4 static	Common bus	16 pin	450 ns MAX
TMS 4050	4096 x 1 dynamic	Common bus	18 pin	300 ns MAX
TMS 4051	4096 x 1 dynamic	Dedicated bus	18 pin	300 ns MAX
TMS 4060	4096 x 1 dynamic	Dedicated bus	22 pin	300 ns MAX
TMS 4070	16384 x 1 dynamic	Dedicated bus	16 pin	300 ns MAX
		ROMS/PROMS		
SN74S371	256 x 8 ROM		20 pin	70 ns MAX
SN74S471	256 x 8 PROM		20 pin	70 ns MAX
SN74S472	512 x 8 PROM		20 pin	55 ns TYP
TMS 4700	1024 x 8 ROM		24 pin	450 ns MAX
TMS 4732	4096 x 8 ROM		24 pin	450 ns MAX
TMS 2708	1024 x 8 EPROM		24 pin	450 ns MAX
TMS 27L08	1024 x 8 EPROM (LOW POWER)		24 pin	450 ns MAX
TMS 2716	2048 x 8 EPROM		24 pin	450 ns MAX
		PERIPHERALS		
TMS 9901	Programmable System Interface		40 pin	
TMS 9902	UART		18 pin	
TMS 9903	USRT		20 pin	
TIM 9904	Clock Generator		20 pin	
TIM 9905	Data multiplexer (SN74LS251)		16 pin	
TIM 9906	Addressable latch (SN74LS259)		16 pin	
TIM 9907	Priority encoder (SN74148)		16 pin	
SN74S412	8-bit I/O port		24 pin	
SN74LS138	3 to 8 Decoder		16 pin	
TMS 6011	UART		40 pin	
SN74S241	Bidirectional bus driver		20 pin	

7. SYSTEM DESIGN EXAMPLES

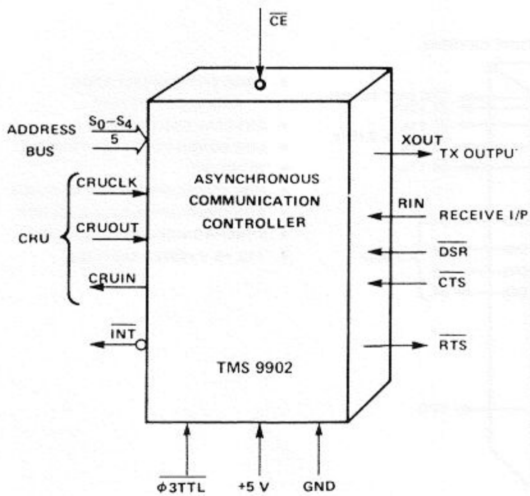
Figure 18 illustrates a typical minimum TMS 9900 system. Eight bits of input and output interface are implemented. The memory system contains 1024 x 16 ROM and 256 x 16 RAM memory blocks. The total package count for this system is 13 packages.

A maximum TMS 9900 microprocessor system is illustrated in Figure 19. ROM and RAM are both shown for a total of 65,536 bytes of memory. The I/O interface supports 4096-output bits and 4096-input bits. Fifteen external interrupts are implemented in the interrupt interface. The clock generator and control section contains memory decode logic, synchronization logic, and the clock electronics. Bus buffers, required for this maximally configured system, are indicated on the system buses.



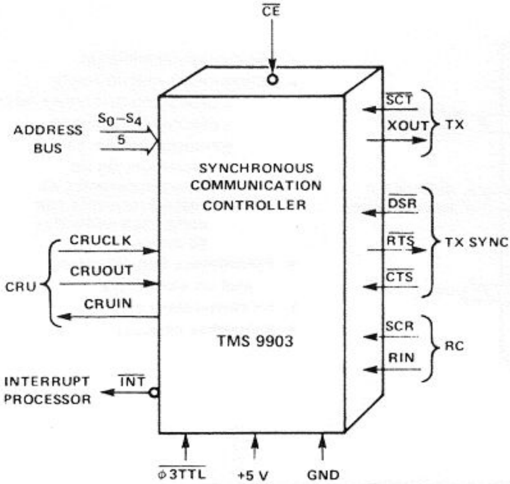
- TMS 9900 CPU PERIPHERAL
- 22 INTERRUPT AND I/O PORTS
  - 6 DEDICATED INTERRUPT INPUTS
  - 7 DEDICATED I/O PORTS
  - 9 PROGRAMMABLE AS INTERRUPT OR I/O
  - INT 3 PROGRAMMABLE AS TIMER INTERRUPT FOR INTERVALS FROM 21 μs TO 349 ms
- EXPANDABLE FOR INTERRUPT AND I/O EXPANSION
- 5 V POWER SUPPLY
- N-CHANNEL PROCESS

FIGURE 14 – TMS 9901 PROGRAMMABLE INTERRUPT AND I/O CONTROLLER



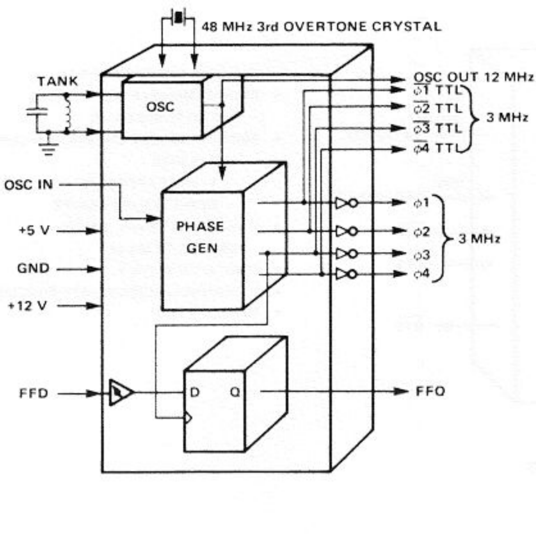
- TMS 9900 CPU PERIPHERAL
- PROGRAMMABLE DATA RATES 110 TO 76,800 BAUD
- PROGRAMMABLE CHARACTER LENGTH
  - 5–8 BITS
  - 1–1½–2 STOP BITS
  - ODD-EVEN-NO PARITY
- ON-CHIP INTERVAL TIMER 64 μs TO 16,384 μs
- SINGLE 5 V SUPPLY
- N-CHANNEL SILICON-GATE PROCESS
- 18 PIN 0.3" DIP

FIGURE 15 – TMS 9902 ASYNCHRONOUS COMMUNICATIONS CONTROLLER



- TMS 9900 CRU PERIPHERAL
- DC TO 250 K BITS/SEC
- PROGRAMMABLE SYNC. REGISTER AND CHARACTER LENGTH
- BI-SYNC AND SDLC COMPATIBLE
- ON-CHIP INTERVAL TIMER 64 μs TO 16,384 μs
- SINGLE 5 V SUPPLY
- N-CHANNEL SILICON-GATE PROCESS
- 20 PIN 0.3" DIP

FIGURE 16 – TMS 9903 SYNCHRONOUS COMMUNICATIONS CONTROLLER



- SINGLE-CHIP OSCILLATOR AND CLOCK DRIVER
- CRYSTAL CONTROLLED
- LOW-POWER SCHOTTKY PROCESS
- 20 PIN DIP
- NON-OVERLAPPING 12 V 4φ CLOCK
- NON-OVERLAPPING 5 V 4φ CLOCK
- SYNCHRONIZED RESET CIRCUIT
- +12, +5 V POWER SUPPLIES

FIGURE 17 – TIM 9904 CLOCK DRIVER

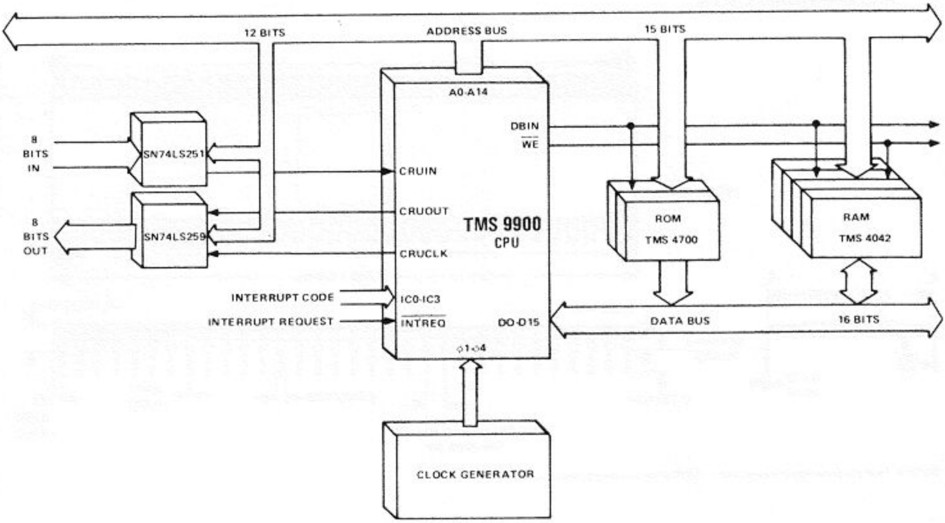


FIGURE 18. – MINIMUM TMS 9900 SYSTEM

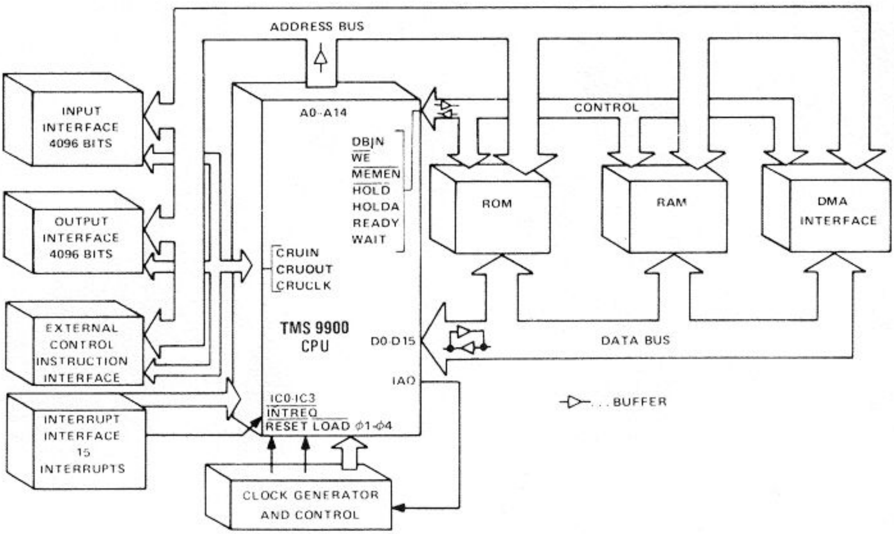
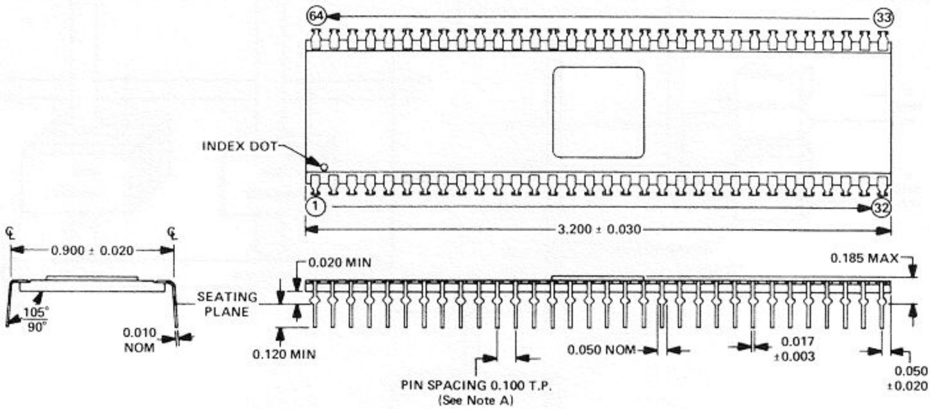


FIGURE 19. – MAXIMUM TMS 9900 SYSTEM

8. MECHANICAL DATA



NOTE A. Each pin centerline is located within 0.010 of its true longitudinal position.

Texas Instruments Northern European Semiconductor Division

UNITED KINGDOM PLANTS AND SALES OFFICES

HEAD OFFICE

Texas Instruments Limited
Northern European
Semiconductor Division,
Manton Lane,
Bedford MK41 7PA
Telephone (0234) 67466
Telex 82178

NESD/TI SUPPLY REGIONAL SALES OFFICES

Bedford
Texas Instruments Limited,
Manton Lane,
Bedford MK41 7PA
Telephone (0234) 67466
Telex 82178

Slough
Texas Instruments Limited,
186 High Street,
Slough, Berks
SL1 1LD
Telephone (0753) 33411
Telex 848363

Manchester
Texas Instruments Limited,
Mersey House,
Battersea Road,
Stockport,
Cheshire SK4 3QR
Telephone (061) 442-7000
(061) 432-0645
Telex 667478

Edinburgh
Texas Instruments Limited,
24 Rutland Street,
Edinburgh,
EH1 2AN
Telephone (031) 229-5573/1481
Telex 72637

Southampton
Texas Instruments Limited,
Carlton House,
Carlton Place,
Southampton SO1 2DZ
Telephone (0703) 27267
Telex 47580

U.K. INDEPENDENT DISTRIBUTORS

Slough
Anzac Components Limited,
396 Bath Road,
Slough, Berks SL1 6JA
Telephone (0753) 38811
Telex 847811

Manchester
Anzac Components Limited,
27 Greek Street,
Stockport,
Cheshire SK3 8AX
Telephone (061) 480-1118
Telex 669664

Sunbury-on-Thames
Cramer Components Limited,
Hawke House,
Green Street,
Sunbury-on-Thames,
Middlesex
Telephone Sunbury 85577
Telex 923592 Cramer G Ldn

U.K. INDEPENDENT DISTRIBUTORS (continued)

Dublin
Hawthorne Electronics Limited,
7 Ontario Terrace,
Dublin.
Telephone (0001) 962022/962308
Telex 31584

Larne
Hawthorne Electronics Limited,
18 Cross Street,
Larne,
Co. Antrim.
Telephone (0574) 5321
Telex 747286

Harlow
ITT Electronic Services,
Edinburgh Way,
Harlow,
Essex.
Telephone (0279) 26777
Telex 817379

Bristol
ITT Electronic Services,
30 Denmark Street,
Bristol.
Telephone (0272) 290125
Telex 449260

Scotland
ITT Electronic Services,
62 High Street,
Johnstone,
Renfrewshire,
Scotland.
Telephone Johnstone 24357
Telex 778622

Manchester
ITT Electronic Services,
110 Corn Exchange,
Manchester M4 3BN
Telephone (061) 832-2681 and
(061) 834-7827/8
Telex 669177

Dublin
ITT Electronic Services,
142 Phibsborough Road,
Phibsborough, Dublin.
Telephone Dublin 307522
Telex 8288

Leicester
ITT Electronic Services,
56 Welford Road,
Leicester.
Telephone (0533) 548292
Telex 341237

Epping
Mogul Electronics Limited,
273 High Street,
Epping,
Essex CM16 4DA
Telephone (0378) 77366
Telex 817326

Derby
Quarndon Electronics (Semiconductors) Limited,
Slack Lane,
Derby DE3 3ED
Telephone (0332) 32651
Telex 37163

Portsmouth
SDS Components Limited,
Jubilee Unit,
Airport Industrial Estate,
Eastern Road,
Portsmouth.
Telephone (0705) 65311
Telex 86114

U.K. INDEPENDENT DISTRIBUTORS (continued)

Scotland
SDS Components Limited,
111 Alexander House,
East Kilbride,
Glasgow G74 1LX.
Telephone (03) 552-48617
Telex 779044

SCANDINAVIAN T.I. PLANTS AND SALES OFFICES

Sweden
Texas Instruments International Trade Corpn.,
(Sverige Filialen),
Fack S100-54,
Stockholm.
Telephone 08-23 54 80
Telex 10377

Norway
Texas Instruments Norway AS,
Ryensvingen 15,
Oslo 6.
Telephone (02) 689485
Telex 19628

Denmark
Texas Instruments A/S,
Marielundvej 46E,
2730 Herlev.
Telephone (02) 91 74 00
Telex 35123

Finland
Texas Instruments Finland OY,
Freesenkatu 6,
P.O. Box 917,
00101 Helsinki 10.
Telephone 40 83 00
Telex 12 1457

SCANDINAVIAN T.I. DISTRIBUTORS

Sweden
AB Gosta Backstrom,
Box 12009,
Alstromergatan 22,
S-102 21 Stockholm 12.
Telephone (08) 541080
Telex 10135

Sweden
Multikomponent,
Fack,
Ankdammsgatan 32,
171 20 Solna.
Telephone (08) 835150
Telex 10516

Multikomponent,
Box 29047,
Lundgatan 12,
400 62 Goteborg.
Telephone (031) 801980

Multikomponent,
Hälsjogatan 4,
217 66 Malmö.
Telephone (040) 130020

Norway
Ingeniorfirma Morgenstierne & Co. AS,
PB 6688 Rodelokka,
Oslo 5.
Telephone (02) 356110
Telex 11719

Printed by Newnorth Artwork Ltd. Bedford England